MASTER'S DEGREE THESIS

Master of Science in Advanced Mathematics and Mathematical Engineering

Title: Non-smooth dynamics in integrate-and-fire neuron models with dynamic threshold

Author: José Eduardo Morales Morales

Advisors: Teresa Martinez-Seara and Gemma Huguet Casades

Department: Matemàtica Aplicada I

Academic year: 2012-2013

Facultat de Matemàtiques i Estadística

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Universitat Politècnica de Catalunya
Facultat de Matemàtiques i Estadística

Treball Final de Màster

Non-smooth dynamics in integrate-and-fire neuron models with dynamic threshold

José Eduardo Morales Morales

Advisors: Maria Teresa Martinez-Seara
Gemma Huguet Casades

Departament de Matemàtica Aplicada I

To my family, professors of the master, and Conacyt MX for their support.

Abstract

Keywords: Computational Neuroscience, Non-Smooth, Dynamical System, Integrate and Fire, Resonate and Fire, Dynamic Threshold.

MSC2000: 200092C20,34C23,34C25,34C60,82C32

The aim of this work is to introduce and study simple neuron models with a dynamic threshold: The integrate and fire with dynamic threshold and a resonate and fire with dynamic threshold. The last model is a new model raised from an initial study of the integrate and fire with a dynamic threshold. The characteristics of the new model are studied from a mathematical point of view specially concerning the dynamical systems theory of the non-smooth systems.

Contents

1. Introduction

There exist many neuroscience models nowadays which are designed for different purposes, some of them model accurately the electrophysiology of the neurons, while others just capture the essence of neurons behavior and which are known as *simple models*. When modeling large networks simple models are preferred due to their computational efficiency [1].
The aim of this work is to introduce and studies simple models with a dynamic threshold: The integrate and fire with dynamic threshold [4] and a resonate and fire with dynamic threshold (class 2 excitability) [3]. The last model is a new model raised from an initial study of the integrate and fire with a dynamic threshold. The characteristics of the new model are studied from a mathematical point of view specially concerning the dynamical systems theory of the non-smooth systems.

As an overall introduction we begin our work including some general information about computational neuroscience in chapter 1. This includes some neuroscience concepts: Electrophysiology of neurons, the Hodgkin Huxley model, and simple models of neurons. This information helped me to familiarize with the concepts and gain insight into the mathematical approach, which already exists in computational neuroscience, and how the theory of dynamical systems plays an important role in it.
On chapter 2 we introduce the integrate and fire model with dynamic threshold. This is a non-linear system with non-smooth characteristics. In particular this system has the purpose to model type 3 excitability [4]. This type of neuron excitability is characterized by the fact that in order for the neuron to fire the input stimulus and rise in voltage must be fast enough. A qualitative and qualitative analysis is performed to the differential equation which includes: The computation of the solutions of the differential equation considering a constant current input and time dependent current input. The expression of the periodic solutions in terms of Fourier series is done and we integrate the equations using numerical computations. A final analysis is considered for the system with constant current in order to forecast the possible number of spikes the neuron model will produce.

The results obtained in chapter 2 led us to propose a new resonator model that we call the *modified resonate and fire model* ($MR\&F$). It is a modification of the original resonate and fire ($R\&F$) introduced in chapter 1. In this modification, the functions of the variables in the $R\&F$ were reassigned and the non-smooth characteristics of the integrate fire model were added. Before, the $R\&F$ model had the variable x as a current-like variable and y as the voltage or membrane potential. Now the $MR\&F$ considers the variable x as voltage and y as the threshold variable. Indeed y can be seen as a current-like variable that opposes to membrane depolarization (for instance as potassium current) raising the threshold to spike. Then, it is natural to associate this current with the threshold. The resetting conditions after an spike are complemented with a Δ_y increment to the threshold variable every time a spike is achieved. These modifications led to a more complex model which still keeps the damped oscillations from the original one and includes threshold variability.

In chapter 3 we deep on the understanding of the $MR\&F$ as a dynamical system depending on two parameters. We describe the possible cases based on the parameters and the characteristics for each case. We introduce a suitable Poincaré map and we look for fixed points, bifurcations and periodic orbits. Both unstable and stable fixed points where found varying the parameters. Their occurrence happens through a saddle-node bifurcation for maps. This is not a typical saddle-node bifurcation due to the non-smooth nature of the system. Period doubling was investigated and a sub-critical pitchfork bifurcation was found for period-2 fixed points. A 3 dimensional fixed points diagram has been constructed in order to gain more insight into the system and the position of the fixed points with respect to the parameters.

Finally the fixed points of the map are linked with the periodic orbits of the system, where the stability of the fixed points gives the stability of the periodic orbits. This can be translated to a sustained periodic spiking, for infinite time or finite time, depending on the initial conditions chosen. The $MR\&F$ model shows a richer dynamics than the $R\&F$. In the last pages of chapter 3 an overall review of the fixed points and the periodic orbits. And a final conclusion of the system studied.

Chapter 1
Neuroscience and Simple Models

1. General Concepts

In Neuroscience behavior of neurons can be perceived as a Nonlinear Dynamical system, they can be modeled in a way that they can explain mathematically how a neuron fires, specifically this is answered through the geometrical theory of dynamical systems. A neuron receives inputs from more than 10,000 neurons through the contacts of its dendritic trees called **synapses**. These inputs produce electrical transmembrane currents that change the membrane potential of the neuron, see fig. 1. Synaptic currents produce changes, which are **called postsynaptic potentials (PSP)**, these are proportional to the currents, (small currents produce small PSP, and visceversa). The neuron has channels in the membrane. These channels produce changes in the voltage of the neuron and work based on some ions, when a current is injected and is produced a PSP. This PSP can be amplified by the voltage sensitive channels and produce an **action potential** or **spike**. A spike is an abrupt and transient change of the membrane voltage that propagates to other neurons through an **axon** (see inset in Fig. 1).

Basic theory of neuroscience models the firing as follows: If the sum of incoming PSPs reaches a certain voltage value called the **firing threshold** then the neuron produces a spike (or action potential). If is below then the neuron remains quiescent. This way of modelling the neurons is in some way very classical, because it has been seen that the **firing threshold** is not well defined in a sense that neurons do not fire exactly after surpassing the firing threshold. Another way to model neurons is by using injected current thresholds instead of voltage thresholds, the current threshold is called **rheobase** (the minimal amplitude of injected current of infinite duration in order to produce an spike or action potential). We can produce PSPs and spikes based on the current that we inject. If we inject an **excitatory input**, and the membrane potential goes closer to the firing threshold, then we say that the neuron **depolarizes**. If we inject an **inhibitory input**, and the membrane potential goes farther of the firing threshold, then we say that the neuron **hyperpolarizes** (i.e. inhibitory input hyperpolarizes the membrane potential).

There are sometimes that hyperpolarization of a neuron produces an spike, even that we moved it away from the firing threshold (see Fig. 2). Basically the behavior

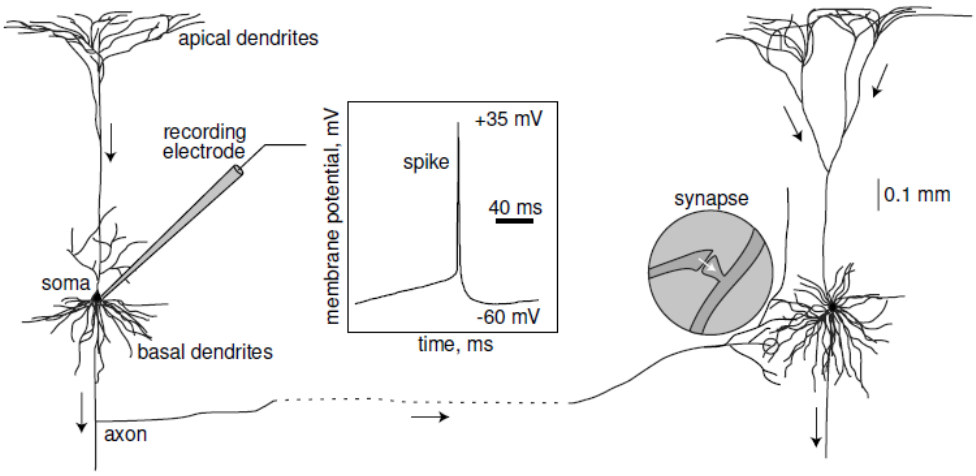

FIG. 1. Connection between neurons. Figure from reference [1].

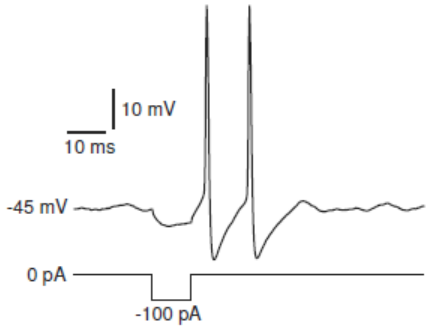

FIG. 2. Rebound spikes in response to a brief hyperpolarization pulse of current. Figure from reference [1].

of a neuron is more complex to model than just stating that a neuron fires when the membrane potential surpasses its firing threshold. Neurons can behave in different ways, they can show just one spike or a lot of spikes by **applying an inhibitory or excitatory current of a given amplitude**, and this behavior is explained by the kind of dynamical system the neuron has indeed and how it behaves with the injected current.

1.1. The Hodgkin Huxley model. The Hodgkin-Huxley formalism is a model to describe the neuronal dynamics in terms of activation and inactivation of voltage-gated conductances. The Hodgkin-Huxley model is a dynamical system. The variables of the system are the membrane potential V, the so so called gating variables n,m and h controlling the persistent potassium (n) and the transient sodium (m,h) currents. The evolution law is given by a *four-dimensional system of ordinary differential equations*. The variables from Hodgkin Huxley describing the neuronal

dynamics can be classified in three classes, according to their function and the time they execute in the mathematical model.

(1) Membrane potential V.
(2) Excitation variables m, such as activation of sodium current (m). These variables are responsible for the upstroke of the spike.
(3) Recovery variables, such as inactivation of a sodium current (h) and activation of a fast potassium current (n). These variables are responsible for the repolarization (downstroke) of the spike.

The excitation variables and recovery variables play an important role in the neuron. As these variables allow the depolarization or repolarization of the membrane potential. In section 2 we do a more detailed introduction of how to arrive to this model and we provide the equations.

1.2. Phase Portraits. Dynamical systems are a powerful tool to understand models in neuroscience. The study of these models will help us to understand the behavior of the neurons. When building a phase portrait for explaining a neuron behavior is usual to do it using a two-dimensional dynamical systems, the membrane potential **V**, and an activation or a recovery variable **n, m, h**). When a Neuron is quiescent (its membrane potential is resting) we interpret this as there are no changes in the state variable of the neuron (membrane potential V and the activation or recovery variables), hence the system is in an **equilibrium point**. If the neuron remain quiescent despite small disturbances and membrane noise, then we conclude the *equilibrium is stable*.

For example, let us consider the membrane potential **V** and the activation variable **n**. The evolution of the model is a trajectory **(V(t), n(t))** on the **V x n plane**. Depending on the initial point, the system can have many trajectories as shown in Figure 3.

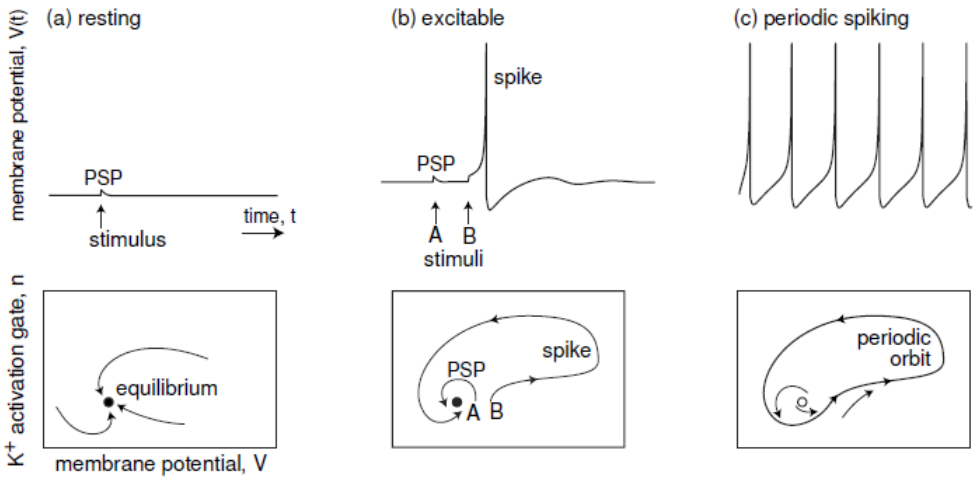

FIG. 3. Different neuron activities. Figure from reference [1].

1.3. Bifurcations. When a dynamical system depending on a parameter μ, X_μ changes its qualitative behaviors at a value μ_0 we say that there is a bifurcation for this value of μ. On the neuron dynamics the bifurcations are present. If we consider the injected current as the parameter to vary, we would have bifurcations of co-dimension 1 and the phase portrait will show the different behaviors of the neuron, from resting to an spiking activity. The existence of equilibrium points and limit cycle attractors are the explanation for the neural activity. Neurons are excitable because they are near bifurcations from resting to spiking activity, and the type of the bifurcation will determine the excitable properties of the neuron.

For the two dimensional models we study we encounter four different bifurcations:

- Saddle-node Bifurcation
- Saddle-node on invariant circle (SNIC) bifurcation
- Subcritical Andronov-Hopf bifurcation
- Supercritical Andronov-Hopf bifurcations

Then the four bifurcations are:

- **Saddle-node Bifurcation** an stable node and a saddle point, when varying the parameter (current) collide and anhilite each other. In the model we study after the saddle node bifurcation the system jumps to a limit cycle attractor which also existed before the bifurcation values (see Fig. 4a). This is interpreted as the neuron starts to fire tonic spikes repeteadly. Because the neuron goes from a resting state to spiking, there should be a coexistance of a limit cycle with the resting state before the bifurcation value in order to make the transition happens.
- **Saddle-node on invariant circle Bifurcation.** Before the bifurcation the two branches of saddle, have an heteroclinic conection with the node, after the bifurcation and the critical point dissapear and the heteroclinic conection becomes a periodic orbit (see Fig. 4b).
- **Subcritical Andronov-Hopf Bifurcation.** A small unstable limit cycle shrinks around an stable equilibrium (focus). And after the bifurcation the equilibrium becomes unstable and the limit cycle dissapears. In the model we study the trajectory diverge from the equilibrium and approaches a large amplitude spiking limit cycle which already existed before the bifurcation value.
- **Supercritical Andronov-Hopf Bifurcation.** An attractive focus becomes unstable and give birth to a small-amplitude limit cycle attractor. In our model when the magnitude of the injected current increase the amplitude of the limit cycle increases an becomes an spiking limit cycle.

The neurons can be classified based on which bifurcation the system undergoes. In the case of the Andronov-Hopf bifurcations, the neurons exhibit damped oscillations of membrane potentials called subtresholds, whereas the saddle-node bifurcations (with on or off an invariant circle) do not. We refer to the neurons with damped subthreshold oscillations as **resonators**, and to the ones who don't have this property as **integrators**. The neurons that exhibit coexistance of resting an spiking states are called **bistable** and the ones who don't as **monostable**. In this case the

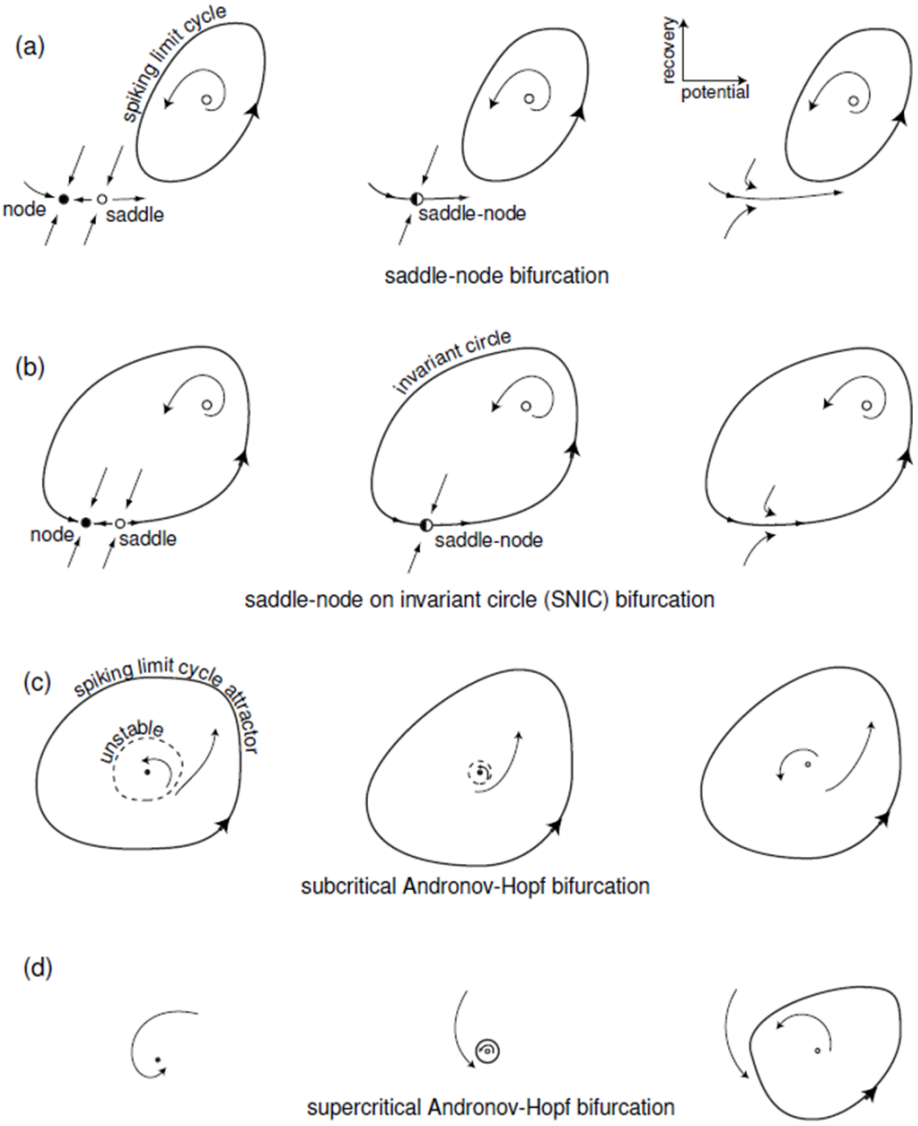

FIG. 4. Four generic bifurcations from the equilibrium state leading to a transition from resting to periodic spiking behaviour in the neurons. Figure from reference [1].

saddle-node and the subcritical Andronov-Hopf bifurcations are bistable, while the other two are monostable.

1.4. Neurocomputational properties.

Considering the previous bifurcations, we proceed to explain how the process of a spike is made and how it reflects in the phase portrait. In this case we assume

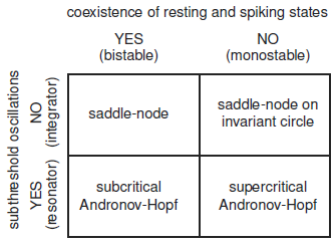

FIG. 5. Classification of the neurons according to their bifurcation state. Figure from reference [1].

that neurons receive shock inputs, (i.e. brief but strong pulses of current that do not change the phase portrait, but only push or reset the state of the neuron into various regions of the phase space). The horizontal axis in Fig. 6 correspond to the membrane potential V with instantaneous sodium current, and the vertical axis corresponds to a recovery variable (can be the potassium current (n) activation).

In the plots the black circles correspond to the node equilibrium point and denotes the neuronal resting state. The attractive limit cycles in the plot correspond to sustained spiking states, which for these plots only exist in the bistable dynamics case (saddle node and subcritical Andronov-Hopf, see Fig. 6 left). The shaded regions are the attraction domain of the limit cycles. The white region are the attraction domain of the equilibrium. In brief, to initiate a spiking, the external input should push the state of the system into the shaded region, and to extinguish spiking the input should push the state back into the white region.

In the case of monostable dynamics (saddle node on invariant circle and supercritical Andronov-Hopf Bifurcations, see Fig. 6 right) there is no spiking limit cycle attractors, so the entire phase space is the attraction domain of the stable equilibrium.

An important difference between integrators and resonators, is that inhibitory inputs **(hyperpolarization)** prevent spiking in integrators, but can promote it in resonators. In the integrators the excitatory inputs push the state of the system toward the shaded region, while inhibitory inputs push it away. In resonators, the excitation and inhibition push the state toward the shaded region, because the region surround the equilibrium and can be reached along any direction.

One thing about integrators is they have well-defined thresholds and resonators do not. The white circles or saddles in the integrator are stable along the vertical direction and unstable along the horizontal. The two trajectories that lead to saddle along the vertical direction are the **separatrices**. In this case the separatrices play the role of the threshold since only those perturbations that push the state of the system beyond them result in a spike. Notice that the threshold is not a point but a tilted curve that spans a range of voltage values.

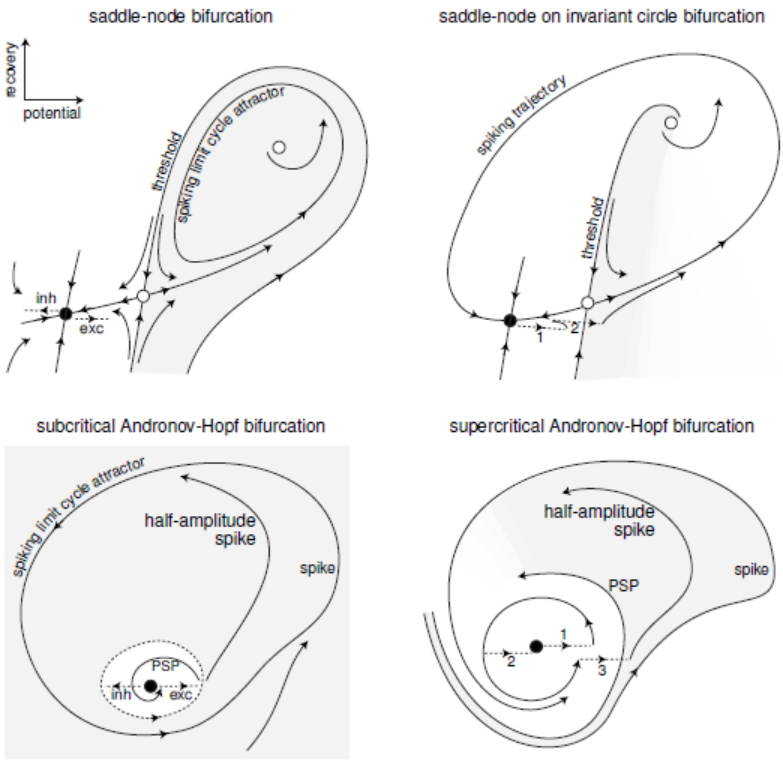

FIG. 6. The geometry of the phase portraits explain many neuro-computational properties. Figure from reference [1].

In the case of the subcritical Andronov-Hopf bifurcation, the threshold is well-defined, (although is a resonator), the threshold is a small unstable limit cycle that separates the attraction domain of stable equilibrium and spiking limit cycle. Just to clarify more in depth the characteristics of integrators vs resonators, the integrators prefer high frequency inputs, the higher the frequency the sooner they fire, so the response of the resonator neuron depends on the frequency content of the input.

2. Electrophysiology of Neurons

2.1. Ions and Nersnt Potential. Electrical activity is sustained and propagated via ionic currents. There exist four ionic species: **sodium (Na^+), potassium(K^+) calcium(Ca^{2+})and chloride(Cl^-)**. The sodium, potassium and calcium are cations because of their positive charge and chloride an anion (negative charge). These chemicals play a relevant role in the neuron. They are inside of the cell as well as outside. The extracellular medium has a high concentration of Na^+ and Cl^- and a relatively high concentration of Ca^{2+}. The intracellular medium has a high concentration of K^+ and negatively charged molecules (denoted by A^-).

These chemicals flow through some gates, depending if the gates are activated or deactivated.

In neuroscience, the **nernst potential** is a value that gives the **equilibrium potential** of the neuron depending of some given interaction of ions (i.e. Na^+ and K^+). For examples, when potassium ions diffuse out of the cell they carry a positive charge and leave a net negative charge inside the neuron. The positive and negative charges accumulate in the opposite sides of the membrane surface creating the **trans-membrane potential** or **membrane voltage**. So after a moment diffusion slows down because potassium ions are attracted to the interior because inside starts to be negatively charged (is filled with A^-), so the potassium go back again until an equilibrium in the charge is achieved. The value of such equilibrium potential is given by the Nersnt equation:

$$E_{ion} = \frac{RT}{zF} \ln \frac{[Ion]_{out}}{[Ion]_{in}} \tag{1}$$

where **R** is the universal gas constant, **T** is temperature, **z** valence of the ion, and **F** faraday constant.

When the membrane potential equals the equilibrium potential, say E_K the net K^+ current, denoted as I_K ($\mu A/cm^2$) is zero. The net K^+ current is proportional to the difference of potential, that is:

$$I_K = g_K(V - E_K) \tag{2}$$

where the positive parameter g_K (mS/cm^2) is the K^+ conductance and $(V - E_K)$ the **driving force**. When the conductance is constant, the current is said to be **Ohmic**. But generally conductance may depend on time and because **time-dependent variation** in conductance is what allows a neuron to generate an action potential.

2.2. An equivalent circuit. The total current **I** flowing across a patch of the cell membrane is the sum of the membrane capacitive current $C\dot{V}$ and all the ionic currents. Using Kirchhoff law of currents we have the differential equations of first order:

$$I = C\frac{dV}{dt} + I_{Na} + I_{Ca} + I_K + I_{Cl} \tag{3}$$

The physical explanation for $\frac{dV}{dt}$ is that it takes time to charge the membrane. The equation substituting every I_{ion}, using equation (2) but for each respective ion is:

$$C\dot{V} = I - g_{Na}(V - E_{Na}) - g_{Ca}(V - E_{Ca}) - g_K(V - E_K) - g_{Cl}(V - E_{Cl}) \tag{4}$$

The membrane potential is typically bounded by the equilibrium potentials in the order:

$$E_K < E_C l < V_{(at\ rest)} < E_{Na} < E_{Ca} \tag{5}$$

Where the inward currents $I_{Na}, I_{Ca} < 0$ increase the membrane potential and make it more positive (**depolarization**) and the outward currents $I_K, I_{Cl} < 0$ make it more negative (**hyperpolarization**).

Most membranes contain a diversity of channels (where the ions go through), for example the Na^+ channels produce an inward current and pull the membrane potential toward the Na^+ equilibrium potential E_{Na}. The value of the membrane potential at which all inward and outward currents balance each other, so the net membrane current is zero, corresponds to the **resting membrane potential**. From equation (4), setting $I = 0$ and $\dot{V} = 0$, we get the equation:

$$V_{rest} = \frac{g_{Na}E_{Na} + g_{Ca}E_{Ca} + g_K E_K + g_{Cl}E_{Cl}}{g_{Na} + g_{Ca} + g_K + g_{Cl}} \tag{6}$$

So the equation (4) can be rewritten as:

$$C\dot{V} = I - g_{inp}(V - V_{rest}) \tag{7}$$

Where $g_{inp} = g_{Na} + g_{Ca} + g_K + g_{Cl}$ is the total **input conductance**, and the quantity $R_{inp} = \frac{1}{g_{inp}}$ is the **input resistance** and measure the asymptotic sensitivity of the membrane potential to injected or intrinsic currents.

From equation (7) we can found that the an equilibrium point is:

$$V = V_{rest} + RI_{inp} \tag{8}$$

Something important to consider is that ionic conductance g_{ion} and input resistance R_{inp} are functions of V and time.

2.3. Conductances. The ionic channels are proteins through which ions flow down their electrochemical gradients. The electrical conductance of individual channels may be controlled by gating particles **(gates)**. These gates are sensitive to:

- Membrane Potential. Example: voltage-gated Na^+ or K^+ channels.
- Intracellular agents. Example: Ca^{2+}-gated K^+ channels.
- Extracellular agents. Example: neurotransmitters and neuromodulators (AMPA, NMDA, or GABA receptors).

The net current generated by a large population of channels is described by:

$$I = \bar{g}p(V - E) \tag{9}$$

where p is the average proportion of channels in the open state, $\bar{g}$ is the **maximal conductance** of the population, and E is the reverse potential of the current i.e the potential at which the current reverses its direction.

2.3.1. *Voltage-Gated Channels.* When the **gates** are sensitive to membrane potential, then the channels are said to be **voltage-gated**. The gates can be of two types: those that *activate* or open the channels, and those that *inactivate* or close them. The probability of an activation gate being in the open state is denoted by the variable m. The probability of an inactivation gate in the open state is denoted by the variable h. The proportion of open channels in a large population is

$$p = m^a + h^b \tag{10}$$

where a is the number of activation gates and b the number of inactivation gates. The channels can be partially activated ($O < m < 1$), activated ($m = 1$) or deactivated ($m = 0$); *inactivated* ($h = 0$); released from inactivation or *deinactivated* ($h = 1$). Some channels do not have inactivation gates ($b = 0$) and they result on *persistent* currents. And channels with both gates give *transient* currents.

2.3.2. *Activation of Persistent Currents.* The dynamics of the activation variable m is described by the first-order ODE:

$$\dot{m} = (m_\infty(V) - m)/\tau(V) \tag{11}$$

where the *activation function* $m_\infty(v)$ and the *time constant* $\tau(V)$ are measured experimentally. In Fig. 7 we show the functions $m_\infty(v)$, $\tau(V)$, which have a sigmoid and uni-modal shape respectively.

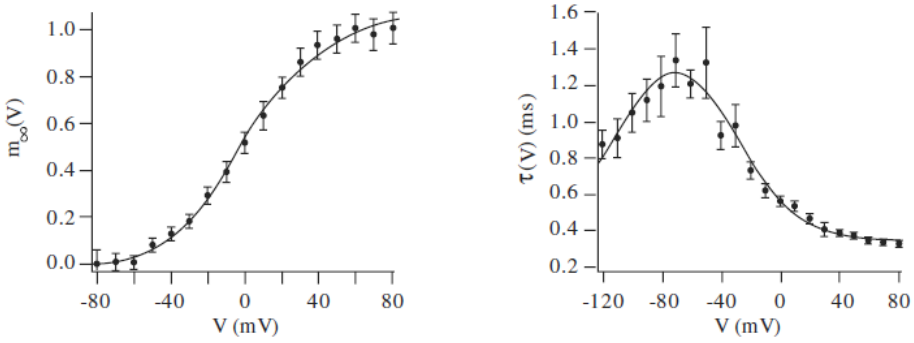

FIG. 7. Activation functions $m_\infty(V)$ and $\tau(V)$. Figure from reference [1].

2.3.3. *Inactivation of Transient Currents.* Similar to m the dynamics of the inactivation variable h is described by the ODE:

$$\dot{h} = (h_\infty(V) - h)/\tau(V), \tag{12}$$

where the *inactivation function* h_∞ is measured experimentally and is depicted in the Fig. 8.

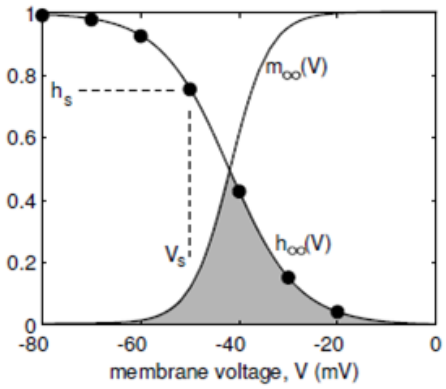

FIG. 8. The steady state activation and inactivation functions overlap (shaded region) and produce a **persistent current**. Figure from reference [1].

2.4. The Hodgkin-Huxley Model. We now proceed to put together what we previously explained, and describe the set of equations that constitutes the **Hodgkin-Huxley Model**. Hodgkin and Huxley determined that the squid axon carries three major currents: voltage-gated persistent K^+ current with four activation gates (resulting in the term n^4 in the equations, where n is the activation variable for K^+); voltage-gated transient Na^+ current with three activation gates and one inactivation gate (the term m^3h), and Ohmic leak current, I_L, which is carried mostly by Cl^- ions. The total set of equations are:

$$C\dot{V} = I - \overbrace{\bar{g}_K n^4 (V - E_K)}^{I_K} - \overbrace{\bar{g}_{Na} m^3 h (V - E_{Na})}^{I_{Na}} - \overbrace{g_L (V - E_L)}^{I_L}$$

$$\begin{aligned}
\dot{n} &= \alpha_n(V)(1-n) - \beta_n(V)n \\
\dot{m} &= \alpha_m(V)(1-m) - \beta_m(V)m \\
\dot{h} &= \alpha_h(V)(1-h) - \beta_h(V)h
\end{aligned}$$

where

$$\begin{aligned}
\alpha_n(V) &= 0.01 \frac{10 - V}{\exp \frac{10-V}{10} - 1} \\
\beta_n(V) &= 0.125 \exp \frac{-V}{80} \\
\alpha_m(V) &= 0.1 \frac{25 - V}{\exp 25 - V 10 - 1} \\
\beta_m(V) &= 4 \exp \frac{-V}{18} \\
\alpha_h(V) &= 0.07 \exp \frac{-V}{20} \\
\beta_h(V) &= \frac{1}{\exp \frac{30-V}{10} + 1}
\end{aligned} \tag{13}$$

The functions $\alpha(V)$ and $\beta(V)$ describe the transition rates between open and closed states of the channels. The equations provided previously are the historical notation, however written in the standard form are:

$$\begin{aligned}
\dot{n} &= (n_\infty(V) - n)/\tau_n(V), \\
\dot{m} &= (m_\infty(V) - m)/\tau_m(V), \\
\dot{h} &= (h_\infty(V) - h)/\tau_h(V),
\end{aligned} \tag{14}$$

where

$$\begin{aligned}
n_\infty &= \alpha_n/(\alpha_n + \beta_n) \qquad & \tau_n &= 1/(\alpha_n + \beta_n), \\
m_\infty &= \alpha_m/(\alpha_m + \beta_m) \qquad & \tau_m &= 1/(\alpha_m + \beta_m), \\
h_\infty &= \alpha_h/(\alpha_h + \beta_h) \qquad & \tau_h &= 1/(\alpha_h + \beta_h),
\end{aligned}$$

The previous functions can be approximated by Boltzmann and Gaussian functions as it is shown in Fig. 9.

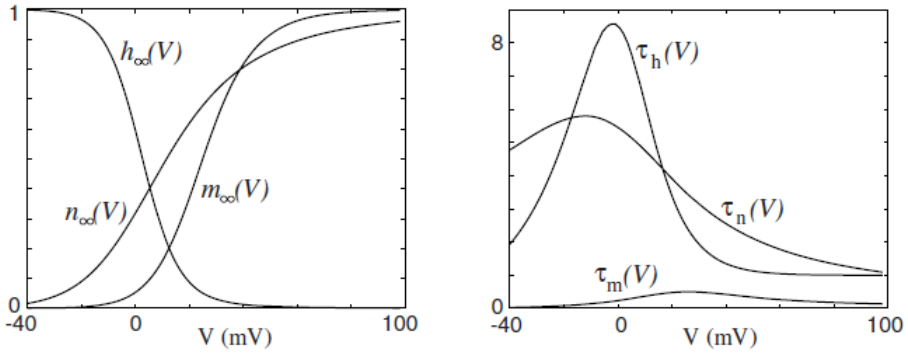

FIG. 9. Steady-state (in)activation functions (left) and voltage-dependent time constants (right) in the Hodgkin-Huxley model. Figure from reference [1].

2.5. Action Potential. We proceed to explain the action potential, in the Fig. 10 we show a typical time course of an action potential in the Hodgkin-Huxley system. The strong depolarization increases activation variables m and n and decreases inactivation variable h. Since $\tau_m(V)$ is small, the variable m is fast. Fast activation of Na^+ conductance drives V toward E_{Na}, resulting in further depolarization and further activation of g_{Na}. This results in an **upstroke** of V. While V moves toward E_{Na}, the slower gating variables catch up. The variable $h \to 0$ creating and inactivation of the Na^+ current and the variable $n \to 0$, causing slow activation of the outward K^+ current. The latter and the leak current repolarize the membrane potential toward V_{rest}.

When V is near V_{rest}, the voltage sensitive time constants $\tau_n(V)$ and $\tau_h(V)$ are relatively large, therefore recovery variables n and h are slow. The outward K^+ current

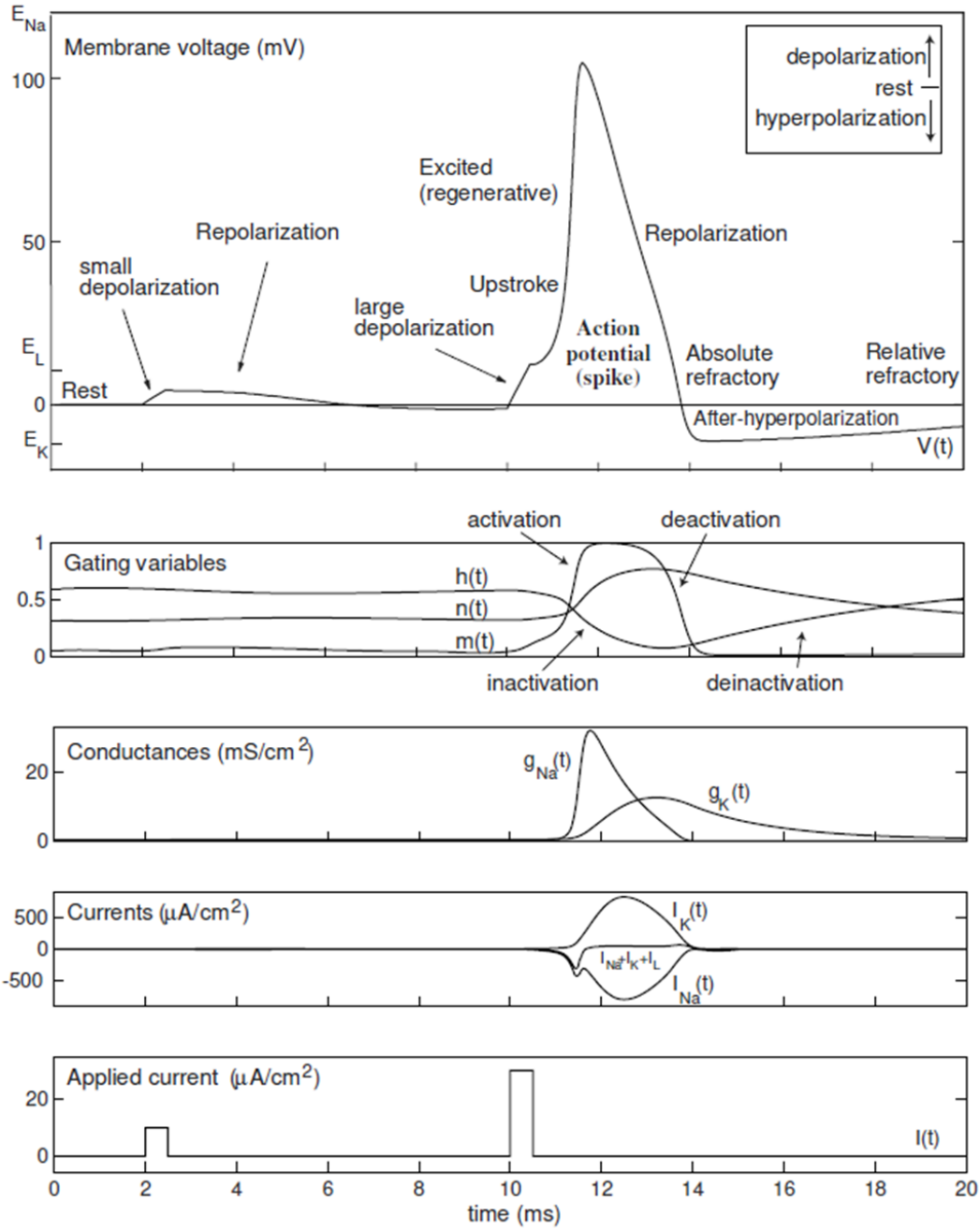

FIG. 10. Time course of several quantatites during an action potential. Figure from reference [1].

continues to be activated (n is large) even after the action potential downstroke,

thereby causing V to go below V_{rest} toward E_K, this is called **afterhyperpolarization**. In addition the Na^+ current continues to be inactivated (h is small) and not available for any regenerative function. The Hodgkin Huxley model cannot generate another action potential during this **absolute refractory** period.

3. Simple Models

Simple models of neurons can reproduce some neurocomputational features of the neuron, as comparison to more complex models like, *Hodgkin-Huxley*, with voltage gated variables. These simple models can help to capture a wide range of properties of different neurons, with a more simplified approach and computational efficiency. Here we present an introduction two different integrate and fire type of models that, we would refer as spiking models. These modelscan be helpful to describing the nature of integrator and resonator neurons.

3.1. Integrate and Fire. The Integrate and Fire is the basic spiking model for an integrator that idealizes and ohmic leakage current and a number of voltage-gated currents. The system can be described by the linear differential equation.

$$C\dot{V} = I - g_{leak}(V - E_{leak}) \tag{15}$$

where g_{leak} refers to the input conductance, and E_{leak} to the equilibrium potential. When V reaches the threshold value E_{thres} the neuron is said to fire an action potential, consequently V is reset to a value E_K (see Fig. 11).

The leaky integrate and fire can be written in the form.

$$\dot{v} = b - v \tag{16}$$

if $v = 1$, then $v \leftarrow 0$. The resting state is $v = b$, the threshold value is $v = 1$, and the reset value $v = 0$. The neuron is excitable when $b < 1$ and fire a periodic spike train when $b > 1$ with a period $T = -\ln(1 - 1/b)$ (see Fig. 12)

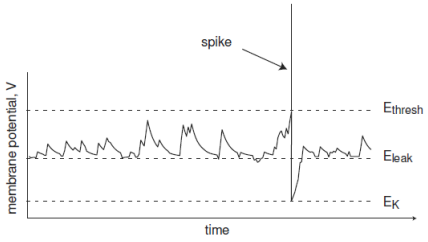

FIG. 11. Integrate and fire model, when $V = E_{thresh}$ the spike is generated and reset to E_K. Figure from reference [1].

Important characteristic of this model to consider are the next:

- **All or none spikes:** All the spikes are equal in size and duration.
- **Well-defined threshold:** The threshold is well defined to a constant value E_{thresh}, and a *spike* is drawed as soon as $V = E_{thresh}$.
- **Relative Refractory Period:** When $E_K < E_{leak}$ the neuron is less excitable immediately after the spike.

- **Excitation and Inhibition** : Excitatory inputs ($I > 0$) will bring the membrane potential closer to the threshold; inhibitory inputs ($I < 0$) do the opposite.

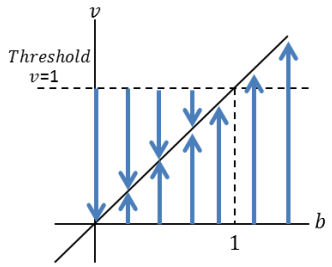

FIG. 12. Integrate and fire model with a phase portrait respect to parameter b.

This simple model for the integrate and fire. Undergoes different bifurcations than the saddle-node bifurcation and the Andronov-Hopf bifurcation (which we previously showed). This happens because the type of bifurcation for this simple model are observed only in piecewise continuous systems.

3.2. Integrate and Fire with periodic time current input. A more complete integrate and fire model has been studied in [2]. It includes a periodic current input. This model is given by the set of equations:

$$\begin{aligned} \frac{dv}{dt} &= -\gamma v + S_0 + S_m \cos(\omega t + \phi) \\ v(t^+) &= 0 \quad if \quad v(t) = v_T \end{aligned} \tag{17}$$

This integrate and fire describes leaky current-clamped membranes in terms of the state variable $v(t)$ (membrane potential), a dissipation constant γ, is an applied constant stimulus S_0, the applied alternating stimulus amplitude S_m, the frequency ω and the phase ϕ respectively. When $v(t)$ reaches a firing threshold at time T then v_T is reset to zero. In other words, the neuron fires and charge again accumulates.

A dimensionless version of the model (17) can be set by the change of variables:

$$u = \frac{v}{v_T} \quad \sigma = \frac{\gamma}{\omega} \quad \S = \frac{S_0}{\omega v_T} \quad B = \frac{S_m}{S_0} \quad \tau = \omega t + \phi$$

Then the system becomes:

$$\begin{aligned} \frac{du}{d\tau} &= -\sigma u + S(1 + B\cos\tau) \\ u(\tau^+) &= 0 \quad if \quad u(\tau) = 1 \end{aligned} \tag{18}$$

This model is analyzed in [2], putting emphasis on his firing phase mapping, where the mapping is done in such a way to force a *firing time* to the next. The study shows that, for some parameter values of the periodic current input, the so called fire dimensional mapping has a finite invariant set, corresponding to a periodic response to periodic forcing, which persists under small changes of parameters. For other parameters values, the firing phases are aperiodic, ergodic or something else. Some important things to consider is that in [2] rotation numbers are computed and they make evidence that *irrational numbers* exist for some parameters of the periodic current input, making evidence that caos can exists for this system based on the parameters values [7].

3.3. Quadratic Integrate and Fire. From the previous integrate and fire model, if we replace from the set of equations (16) the $-v$ with $+v^2$, the system becomes a *quadratic integrate and fire* model.

$$\dot{v} = b + v^2 \tag{19}$$

where if $v = v_{peak}$, then a spike is generated and then $v \leftarrow v_{reset}$. A remark to consider is that v_{peak} is not the threshold, but is a limit for the extension of the spike. From the dynamical system theory the equation (19) is a topological normal form for a *saddle node* bifurcation. Whenever $b > 0$ the right hand side of the model is strictly positive, and the neuron is said to fire a periodic train of action potentials, this is because we reach the peak v_{peak}, and then reset to v_{reset} and we start again. For instance we can define a period for such behavior of spiking that is given by:

$$T = \frac{1}{\sqrt{b}}\left(\arctan\frac{v_{peak}}{\sqrt{b}} - \arctan\frac{v_{reset}}{\sqrt{b}}\right) < \frac{\pi}{\sqrt{b}}. \tag{20}$$

When $b < 0$ we have two zeros $\pm\sqrt{b}$ for the parabola $b + v^2$, where one represents the stable node (resting state) and the other one the unstable node (threshold), see Fig. 13.

The sub-threshold perturbations are those that will push v to go to the resting state (stable node). The super-threshold oscillations will push v beyond the unstable node and will produce the initiation of an action potential which will reach the v_{peak} and then resetting v to v_{reset}. It is possible to produce different kind of behaviors based on the position of v_{reset} and the b parameter. In Figure 14 a complete overview of how we can have an excitable neuron or a periodic spiking neuron based on the bifurcation parameter b and the value v_{reset} is given.

We can conclude that the quadratic integrate and fire neuron is an integrator, exhibits saddle-node bifurcation, has a (dynamic) threshold and it generates spikes with latencies. This model can be used in simulations of large-scale networks of integrators.

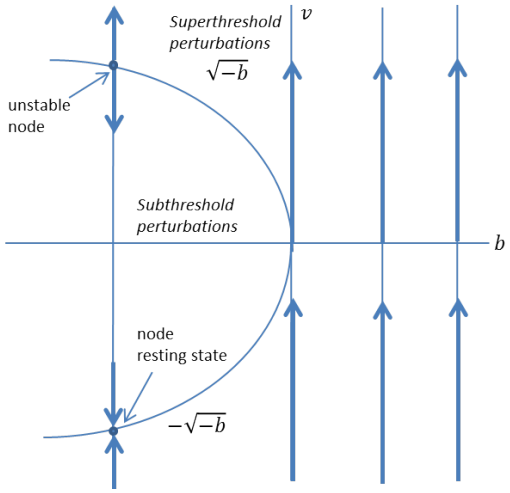

FIG. 13. Bifurcation saddle-node for quadratic integrate and fire.

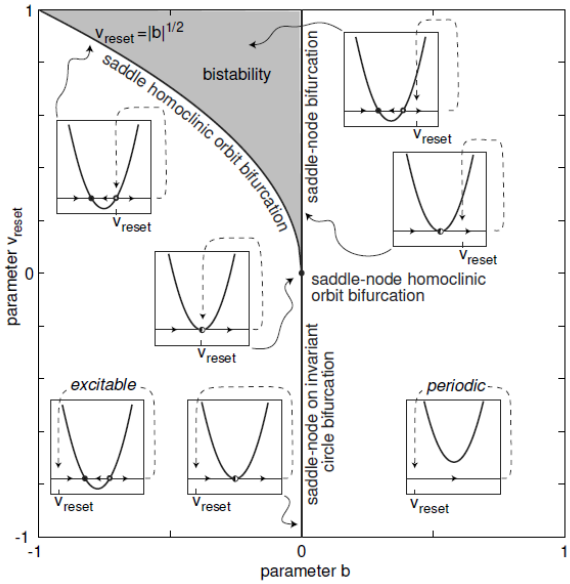

FIG. 14. Different states performed with the parameter v_{reset} and b. Figure from reference [1].

3.4. Resonate and Fire Model. The resonate and fire model from [3] is an extension of the integrate and fire model. Its intention is to model neurons with damped or sustained sub-threshold oscillations of membrane potential and which are observed in many biological neurons. The resonate and fire model illustrates how sub-threshold oscillations may affect a neuron's spiking dynamics. It is also a simple model which exhibits damped oscillations in membrane potential. Its computational efficiency is comparable with the integrate-fire model which makes

it suitable for simulations of large networks of spiking neurons. An example of damped oscillations is shown in Figure 15.

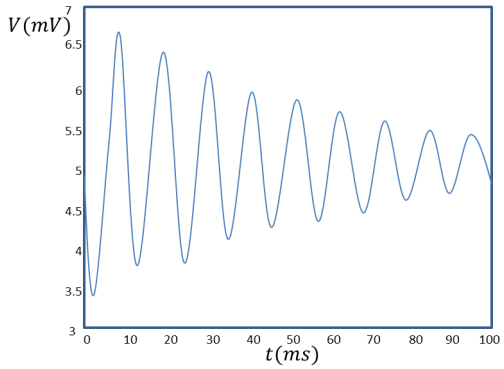

FIG. 15. Damped subthreshold oscillations of membrane potential.

The set of equations for the resonate and fire is given by:

$$\begin{aligned} \dot{x} &= bx - \omega y \\ \dot{y} &= \omega x + by \end{aligned} \tag{21}$$

which in complex form can be recast as:

$$\dot{z} = (bx + i\omega)\, z \tag{22}$$

Here $z = x + iy \in \mathbb{C}$ is a complex valued variable which describes the oscillatory activity of the neuron. The real part x belongs to the current-like variable, and the imaginary part y is the voltage-like variable. $b + i\omega \in \mathbb{C}$ is a parameter where $b < 0$ is the rate of attraction to the rest and $\omega > 0$ is the frequency of the oscillation. This is the most simple model showing damping oscillations and describes sub-treshold behavior of a neuron.
In this model we say that z fires an action potential when the voltage variable $y = \imath z$ crosses the threshold value $y = 1$ which is a horizontal line that passes through $i \in \mathbb{C}$ (see Fig. 16). To make it fire external perturbations should push z beyond the curve that surrounds the white area in the Figure 16. The curve is a piece of solution that passes through the point $i \in \mathbb{C}$ and is tangent to the threshold line. Any solution starting in the white area will converge to the rest state $z = 0$ without crossing the threshold. Any solution starting outside the white area will cross the threshold before converging to the rest state.

Some additional properties of the system is that resonators can easily fire an spike in response to inhibitory stimulation, whereas integrate and fire neurons do not exhibit such neuro-computational properties. Another property for this model is when a sequence of pulses are performed, maybe each pulse cannot evoke an action

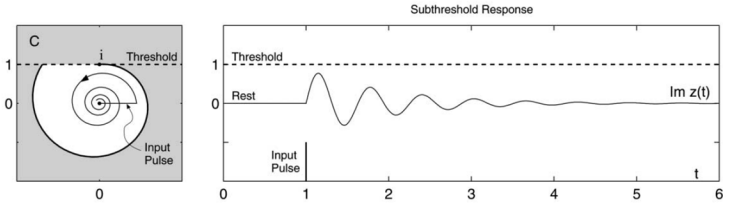

FIG. 16. Threshold is $y = 1$. Figure from reference [3].

potential but together can do it if they have the appropriate timing. For example a first pulse can evoke a damped oscillation, while a second pulse can push the solution beyond the threshold depending on its timing relative to the phase of the oscillation (see Fig. 17 a).

The neuron oscillates with the period $T = 2\pi/\omega$. When the interval between pulses interval is infinitesimal (Fig. 17 (a)) or near the period T (as in 17 (b)) then the neuron fires. If the interval between pulses is significantly less or greater than the period T, the neuron may not fire. When the interval between pulses are equal to the period then we say the spikes are *resonant*.

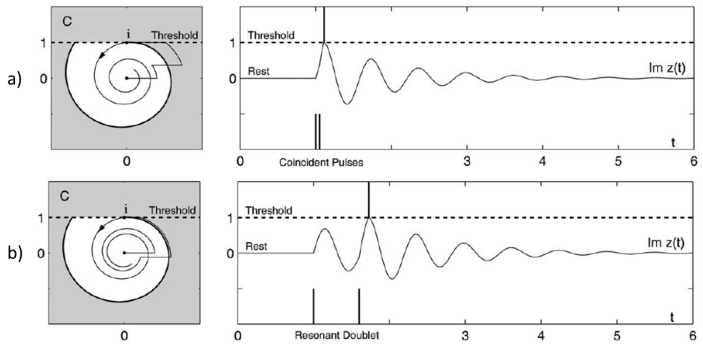

FIG. 17. Different spike behaviors based on stimulation. Figure from reference [3].

Another characteristic is that this system can show bistability between rest and spiking states. For example, if the reset value $i \in \mathbb{C}$ lies outside the basin of attraction of the rest state z_0, as soon as the neuron fires, an spike will keep firing periodically until some perturbation pushes the solution inside the white area which is the basin of attraction of the rest state.

The resonate-and-fire model illustrates the most important features of resonators: damped oscillations, frequency preference, post-inhibitory (rebound) spikes, it can have sustained subtreshold oscillations of membrane potential. This model complements the integrate and fire neurons. Although these simple models are difficult to treat analytically, they can be used when large-scale simulations are concerned.

Chapter 2
Integrate and Fire with dynamic threshold.

1. The model

There exists different type of excitability on neurons based on the way they response to the current input. For the neurons with type 1 excitability is when the input is applied, the transition involves the appearance of a stable limit cycle from a saddle-node on invariant circle. Neurons with type 2 is when the steady state loses stability and repetitive firing emerges from a Hopf-bifurcation. There exists a type 3 excitability which do not involve a bifurcation from a fix point to a periodic orbit when a current input is applied. They do not show repetitive firing for steady currents. For time-varying inputs they fire once if the input rise is fast enough. This type 3 excitability neurons is seen for instance in the auditory brainstem and the kind of models are known as *differentiators* or *slope detectors*, where they can encode the occurrence and time or rapid change in the stimulus. In order for a phasic neuron to fire the input and the rise in voltage must be fast enough, where slow inputs prevents the neuron from reaching the spike threshold [4]. In this section we present the $V-\theta$ model from [4] which is an idealized model for a Type 3 excitability in the form of a leaky integrate and fire model with a dynamic θ-threshold. There exist other models with dynamic threshold however the characteristic of this model is that has the property that the dynamic threshold θ can rise with the sub-threshold voltage V.

The $V-\theta$ model is described by the system of equations:

$$\begin{aligned} \frac{dV}{dt} &= -V + V_r + I(t) \\ \frac{d\theta}{dt} &= -\frac{\theta - f(V)}{\tau_\theta} \end{aligned} \tag{23}$$

where,

$$f(V) = a + e^{(b(V-c))}, \tag{24}$$

is the steady state value of the threshold θ at a voltage V; τ_θ is a time constant for the threshold; V_r is the voltage at the resting state and $I(t)$ is the injected current.

As this system can be described a leaky Integrate and Fire model, the resetting conditions after a spike are:

$$\text{If } V(t^-) = \theta(t^-) \quad \Rightarrow \begin{cases} V(t^+) = V_{reset} \\ \theta(t^+) = \theta(t^+) + \Delta_\theta \end{cases} \tag{25}$$

where Δ_θ is constant.

It is evident the non-smooth nature of this system due to the conditions previously stated.

1.1. Analysis of the $V - \theta$ system.

We consider model (23) and (24) with both constant or time dependent input $I(t)$. We proceed to understand the $V - \theta$ system (23), which includes the approach through:

- Study of solutions with a constant or time dependent current input of the differential equation (23).
- Understand the qualitative properties of the dynamical system (23) (24)
- Perform a numerical integration of the model with Fourier series for the time dependent current case to find the periodic solution of equation (23)
- Understand the dynamics of the non-smooth system (23) (24) with constant current input.

the study of the non-smooth dynamical system with a periodic current input will be left as a future work.

2. Quantitative and Qualitative Analysis of the differential equations.

2.1. Solution of the differential equation with constant current input.

Regular solutions of the $V - \theta$ model are computed analytically discarding for the moment the non-smooth characteristics. We follow different approaches in order to get an overall insight of the solutions. When considering a constant current for the $V - \theta$ model (23), as this is a first order system our solution for the system can be found through the variational constant method for O.D.E.

For V we have the system:

$$\begin{aligned} \frac{dV}{dt} &= -V + V_r + I \\ \frac{d\theta}{dt} &= -\frac{\theta - f(V)}{\tau_\theta} \end{aligned} \tag{26}$$

where their solution for V is composed of an homogeneous and a particular solution given by:

$$V_{h(t)} = ke^{-t} \quad V_p = c$$

where k is a constant. The particular solution for V is given by the constant c, so we have that :

$$0 = -c + V_r + I \quad \Rightarrow \quad c = V_r + I$$

so we have that:

$$V(t) = V_h(t) + V_p = ke^{-t} + Vr + I$$

where the initial condition is $V(t_0) = V_0$ and we compute the value for k:

$$V(t_0) = V_0 = ke^{-t_0} + V_r + I \quad \Rightarrow \quad k = \frac{V_0 - V_r - I}{e^{-t_0}} = e^{t_0}(V_0 - V_r - I)$$

if we set $t_0 = 0$ this give us the solution for V(t) as:

$$V(t) = e^{-t}(V_0 - V_r - I) + V_r + I. \tag{27}$$

In the case for θ, our respective equation from (26) we express it as:

$$\dot{\theta} = -\frac{\theta}{\tau_\theta} + \frac{f(V)}{\tau_\theta} = -\frac{\theta}{\tau_\theta} + \frac{1}{\tau_\theta}\left(a + e^{b(V(t)-c)}\right). \tag{28}$$

For finding the solution for θ we follow the variational constant method, so we consider a solution for this equation of the type:

$$\theta(t) = k(t)e^{-\frac{t}{\tau_\theta}}$$

differentiating we get:

$$\dot{\theta} = -\frac{1}{\tau_\theta}k(t)e^{-\frac{1}{\tau_\theta}t} + k'(t)e^{-\frac{1}{\tau_\theta}t}$$

which gives:

$$k'(t)e^{-\frac{1}{\tau_\theta}t} = \frac{1}{\tau_\theta}\left(a + e^{b(V(t)-c)}\right)$$

$$k'(t) = \frac{e^{\frac{t}{\tau_\theta}}}{\tau_\theta}\left(a + e^{b(V(t)-c)}\right) \tag{29}$$

and integrating $k'(t)$ from t_0 to t we have that:

$$k(t) = k_0 + \int_{t_0}^{t} \frac{e^{\frac{s}{\tau_\theta}}}{\tau_\theta}\left(a + e^{b(V(s)-c)}\right) ds$$

So θ can be rewritten as:

$$\theta(t) = e^{-\frac{t}{\tau_\theta}}\left\{k_0 + \int_{t_0}^{t} \frac{e^{\frac{s}{\tau_\theta}}}{\tau_\theta}\left(a + e^{b(V(s)-c)}\right) ds\right\} \tag{30}$$

where the initial condition is $\theta(t_0) = \theta_0$, and we compute the value for k_0 from the previous expression:

$$\theta(t_0) = \theta_0 = e^{-\frac{t_0}{\tau_\theta}} k_0 \quad \Rightarrow \quad k_0 = \theta_0 e^{\frac{t_0}{\tau_\theta}}$$

Putting all together in (30), and considering $t_0 = 0$ we have our defined solution for θ:

$$\theta(t) = \theta_0 e^{\frac{-t}{\tau_\theta}} + \int_{0}^{t} \frac{e^{\frac{s-t}{\tau_\theta}}}{\tau_\theta}\left(a + e^{b(V(s)-c)}\right) ds. \tag{31}$$

2.2. Solution of the differential equation with periodic time dependent current input.

We have deduced a solution for the $V - \theta$ model considering a constant current input, however in this section we compute a solution considering a time dependent current input $I(t)$.

From (23) we have that:

$$\frac{dV}{dt} = -V + V_r + I(t) \tag{32}$$

where a suitable solution for V can be from the variational constant method, that is consider the solution type:

$$V(t) = k(t)e^{-t} \tag{33}$$

differentiating we get:

$$\dot{V}(t) = -k(t)e^{-t} + k'(t)e^{-t}$$

which gives:

$$k'(t)e^{-t} = V_r + I(t) \quad \Rightarrow \quad k'(t) = e^t\left(V_r + I(t)\right)$$

and integrating $k'(t)$ from t_0 to t we have that:

$$k(t) = k_0 + \int_{t_0}^{t} e^s\left(V_r + I(s)\right) ds$$

and (33) is re-expressed as:

$$V(t) = e^{-t}\left\{k_0 + \int_{t_0}^{t} e^s\left(V_r + I(s)\right) ds\right\} \tag{34}$$

where the initial condition is $V(t_0) = V_0$, so we compute the value for k_0 from (34)

$$V(t_0) = V_0 = k_0 e^{-t_0} \quad \Rightarrow \quad k_0 = V_0 e^{t_0}$$

putting all together in (34) we get

$$V(t) = V_0 e^{t_0 - t} + \int_{t_0}^{t} e^{s-t}\left(V_r + I(t)\right) ds \tag{35}$$

2.3. The periodic solution of the differential equation as a Fourier Series.

In this section we compute a periodic solution in terms of Fourier Series for (23). The resulting series where coded on Matlab. The respective code can be found in the Appendix section.

2.3.1. *Solution of V in Fourier Series.* We proceed to calculate a solution for V for the system (23), where we consider an input signal of the type:

$$I(t) = A[\sin(2\pi\omega t)]^+ \tag{36}$$

where A is the amplitude, and $[\sin(s)]^+ = \max(\sin(s), 0)$, in other words is a *half-wave rectified sinusoid signal.*

Recall from (23) the equation for $\dot{V}$ considering this I(t) is:

$$\dot{V}(t) = -V + V_r + A[\sin(2\pi\omega t)]^+. \tag{37}$$

We can define a solution for (37) for the homogeneous case:

$$V_h(t) = ke^{-(t-t_0)}. \tag{38}$$

And a particular solution V_p in simple Fourier terms:

$$V_p(t) = \frac{\bar{a}_0}{2} + \sum_{k=1}^{n} \bar{a}_k \cos\left(\frac{2\pi}{T}kt\right) + \sum_{k=1}^{n} \bar{b}_k \sin\left(\frac{2\pi}{T}kt\right). \tag{39}$$

So our complete solution is given by:

$$V(t) = V_h(t) + V_p(t) = ke^{-(t-t_0)} + V_p(t)$$

with initial conditions $V(t_0) = V_0$. So we have:

$$V(t_0) = V_0 = k + V_p(t_0) \Rightarrow k = V_0 - V_p(t_0)$$

and the solution considering the initial condition is:

$$V(t) = (V_0 - V_p(t_0))e^{-(t-t_0)} + V_p(t). \tag{40}$$

Notice that $I(t) = A[\sin(2\pi\omega t)]^+$ has period $T = \frac{1}{\omega}$, that is:

$$I(t + \tfrac{1}{\omega}) = A[\sin(2\pi\omega(t + \tfrac{1}{\omega}))]^+ = A[\sin(2\pi\omega t + 2\pi)]^+ = A[\sin(2\pi\omega t)]^+.$$

Let us define $f(t) = V_r + A[\sin(2\pi\omega t)]^+$, then (37) can be expressed as:

$$\dot{V} = -V + V_r + [A\sin(2\pi\omega t)]^+ = -V + f(t) \tag{41}$$

let us express $f(t)$ as a Fourier series:

$$f(t) = V_r + [A\sin(2\pi\omega t)]^+ = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos\left(\frac{2\pi}{T}kt\right) + \sum_{k=1}^{n} b_k \sin\left(\frac{2\pi}{T}kt\right) \tag{42}$$

where the coefficients are given by:

$$a_0 = \frac{2}{T}\int_0^T f(t)dt = 2\omega \int_0^{\frac{1}{\omega}} V_r + A[\sin(2\pi\omega t)]^+ dt \tag{43}$$

(44)
$$a_k = \frac{2}{T}\int_0^T f(t)\cos\left(\frac{2\pi kt}{T}\right)dt = 2\omega \int_0^{\frac{1}{\omega}} \left(V_r + A[\sin(2\pi\omega t)]^+\right)\cos\left(2\pi\omega kt\right)dt$$

(45)
$$b_k = \frac{2}{T}\int_0^T f(t)\sin\left(\frac{2\pi kt}{T}\right)dt = 2\omega \int_0^{\frac{1}{\omega}} \left(V_r + A[\sin(2\pi\omega t)]^+\right)\sin\left(2\pi\omega kt\right)dt$$

Differentiating the particular solution $V_p(t)$ in (39) we obtain:

$$\dot{V}_p = \sum_{k=1}^{n} -\frac{2\pi}{T} k\bar{a}_k \sin\left(\frac{2\pi}{T}kt\right) + \frac{2\pi}{T}\sum_{k=1}^{n} k\bar{b}_k \cos\left(\frac{2\pi}{T}kt\right). \tag{46}$$

Using that (39), (46) and (42) should satisfy (39) we have:

$$\sum_{k=1}^{n} -\frac{2\pi}{T} k\bar{a}_k \sin\left(\frac{2\pi}{T}kt\right) + \frac{2\pi}{T}\sum_{k=1}^{n} k\bar{b}_k \cos\left(\frac{2\pi}{T}kt\right) =$$

$$-\left(\frac{\bar{a}_0}{2} + \sum_{k=1}^{n} \bar{a}_k \cos\left(\frac{2\pi}{T}kt\right) + \sum_{k=1}^{n} \bar{b}_k \sin\left(\frac{2\pi}{T}kt\right)\right) +$$

$$\frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos\left(\frac{2\pi}{T}kt\right) + \sum_{k=1}^{n} b_k \sin\left(\frac{2\pi}{T}kt\right)$$

$$= -\frac{\bar{a}_0}{2} + \frac{a_0}{2} + \sum_{k=1}^{n} (a_k - \bar{a}_k)\cos\left(\frac{2\pi}{T}kt\right) + \sum_{k=1}^{n} (b_k - \bar{b}_k)\sin\left(\frac{2\pi}{T}kt\right)$$

From the previous expression we can obtain the coefficients $\bar{a}_0, \bar{a}_k, \bar{b}_k$, that is:

$$\begin{aligned} \bar{a}_0 &= a_0 \\ -\frac{2\pi}{T}k\bar{a}_k = b_k - \bar{b}_k \Rightarrow &\quad \bar{b}_k = b_k + \frac{2\pi}{T}k\bar{a}_k \\ \frac{2\pi}{T}k\bar{b}_k = -\bar{a}_k + a_k \Rightarrow &\quad \bar{a}_k = a_k - \frac{2\pi}{T}k\bar{b}_k \end{aligned} \tag{47}$$

where solving the system (47) we get:

$$\begin{aligned} \bar{b}_k &= \left(\frac{T^2}{T^2 + 4\pi k^2}\right)\left(b_k + \frac{2\pi k}{T}a_k\right) = \left(\frac{1}{1 + 4\pi k^2\omega^2}\right)(b_k + 2\pi k\omega a_k) \\ \bar{a}_k &= \left(\frac{T^2}{T^2 + 4\pi k^2}\right)\left(a_k - \frac{2\pi k}{T}b_k\right) = \left(\frac{1}{1 + 4\pi k^2\omega^2}\right)(a_k - 2\pi k\omega b_k) \end{aligned} \tag{48}$$

where the coefficients $\bar{a}_0, \bar{a}_k, \bar{b}_k$ are in terms of a_0, a_k, b_k, defined in (43),(44),(45), respectively. We calculate their integrals, getting for a_0:

(49)

$$a_0 = 2\omega \int_0^{\frac{1}{\omega}} V_r + A[\sin(2\pi\omega t)]^+ dt = 2\omega\left(\int_0^{\frac{1}{\omega}} V_r dt + \int_0^{\frac{1}{2\omega}} A\sin(2\pi\omega t)dt\right) = 2\left(V_r + \frac{A}{\pi}\right).$$

For a_k, $k \geq 1$ we have:

$$\begin{aligned} a_k &= 2\omega \int_0^{\frac{1}{\omega}} \left(V_r + A[\sin 2\pi\omega t]^+\right) \cos\left(2\pi\omega k t\right) dt \\ &= 2\omega \left(\int_0^{\frac{1}{\omega}} V_r \cos(2\pi\omega k t) dt + \int_0^{\frac{1}{2\omega}} A \sin(2\pi\omega t) \cos(2\pi\omega k t) dt \right) \\ &= \frac{A((-1)^k + 1)}{\pi(1-k^2)}. \end{aligned} \tag{50}$$

Evaluating the previous integral for $k = 1$ we get that $a_1 = 0$. So the overall values for a_k, $k \geq 1$ are:

$$a_k \rightarrow \begin{cases} \frac{2A}{\pi}\left(\frac{1}{1-k^2}\right) \text{ for k even} \\ \\ 0 \text{ for k odd} \end{cases} \tag{51}$$

For b_k, $k \geq 2$:

$$\begin{aligned} b_k &= 2\omega \int_0^{\frac{1}{\omega}} \left(V_r + A[\sin\left(2\pi\omega t\right)]^+\right) \sin\left(2\pi\omega k t\right) dt \\ &= 2\omega \left(\int_0^{\frac{1}{\omega}} V_r \sin(2\pi\omega k t) dt + A \int_0^{\frac{1}{2\omega}} \sin(2\pi\omega t) \sin(2\pi\omega k t) dt \right) \\ &= 0 \end{aligned} \tag{52}$$

and for b_1 we get that:

$$b_1 = 2\omega \left(\int_0^{\frac{1}{\omega}} V_r \sin(2\pi\omega t) dt + A \int_0^{\frac{1}{2\omega}} \sin(2\pi\omega t) \sin(2\pi\omega t) dt \right) = \frac{A}{2}. \tag{53}$$

So the values for b_k are:

$$b_k \rightarrow \begin{cases} \frac{A}{2} \text{ for k=1} \\ \\ 0 \text{ otherwise} \end{cases} \tag{54}$$

Rewriting (42) with the coefficients and considering $k = 2k$ because a_k takes nonzero values only for even k:

$$f(t) = Vr + \frac{A}{\pi} + \frac{A}{2}\sin\left(\frac{2\pi}{T}t\right) + \sum_{k=1}^{\infty} \frac{2A}{\pi}\left(\frac{1}{1-4k^2}\right)\cos\left(\frac{4\pi k}{T}t\right). \tag{55}$$

Now we will express the particular solution V_p. Notice that the coefficients $\bar{a}_k$ and $\bar{b}_k$ are given in terms of a_o, a_k, b_k as it can be seen on (48).

If we look closely to each value of $\bar{a}_k$ for k=1,2,3,4 we have:

$$\bar{a}_1 = \left(\frac{1}{1+4\pi\omega^2}\right)(b_1 + 2\pi\omega a_1) = \left(\frac{1}{1+4\pi\omega^2}\right)(-\pi\omega A)$$
$$\bar{a}_2 = \left(\frac{1}{1+16\pi\omega^2}\right)(b_k + 4\pi\omega a_k) = \left(\frac{1}{1+16\pi\omega^2}\right)\left(\frac{2A}{\pi}\left(\frac{1}{-3}\right)\right)$$
$$\bar{a}_3 = 0$$
$$\bar{a}_4 = \left(\frac{1}{1+64\pi\omega^2}\right)(b_k + 8\pi\omega a_k) = \left(\frac{1}{1+64\pi\omega^2}\right)\left(\frac{2A}{\pi}\left(\frac{1}{-15}\right)\right)$$

So is clear that $\bar{a}_k$ have particular values for $k = 1$ and k even, while for k odd the values are zero.

If we do the same for $\bar{b}_k$ we get:

$$\begin{aligned} &\bar{b}_1 = \frac{A}{2}\left(\frac{1}{1+4\pi\omega^2}\right) \\ &\bar{b}_k = 4k\omega A\left(\frac{1}{1-k^2}\right)\left(\frac{1}{1+4\pi k^2\omega^2}\right) \text{ for } k\text{even.} \\ &\bar{b}_k = 0\text{for } k\text{odd.} \end{aligned} \tag{56}$$

Now with the coefficients specified, we express the particular solution V_p considering $k = 2k$ for $\bar{a}_k$ and $\bar{b}_k$. That is because they take nonzero values only for k even, while their only nonzero value for k odd is when $k = 1$.
And the final particular solution V_p expressed as Fourier series is:

$$\begin{aligned} V_p(t) = &\overbrace{Vr + \frac{A}{\pi}}^{\bar{a}_0} + \overbrace{\frac{1}{1+4\pi\omega^2}(-\pi\omega A)}^{\bar{a}_1}\cos(2\pi\omega t) + \overbrace{\frac{1}{1+4\pi\omega^2}\left(\frac{A}{2}\right)}^{\bar{b}_1}\sin(2\pi\omega t) \\ &+ \sum_{k=1}^{\infty}\overbrace{\frac{1}{1+16\pi k^2\omega^2}\left(\frac{2A}{\pi}\left(\frac{1}{1-4k^2}\right)\right)}^{\bar{a}_k}\cos(4k\pi\omega t) \\ &+ \sum_{k=1}^{\infty}\overbrace{\frac{1}{1+16\pi k^2\omega^2}\left(8Ak\omega\left(\frac{1}{1-4k^2}\right)\right)}^{\bar{b}_k}\sin(4k\pi\omega t) \end{aligned} \tag{57}$$

and we have defined the particular solution for V(t).

2.3.2. **Solution of θ in Fourier Series.** Once we have calculated V, the term θ depends nonlinearly on V. Therefore, the coefficients of the Fourier series for a θ solution will have a complicated expression, and their values will have to be computed numerically for the whole period $T = \frac{1}{\omega}$. This process is explained in this section.

Let us recall the expression for $\dot{\theta(t)}$ from (26):

$$\frac{d\theta}{dt} = -\frac{\theta}{\tau_\theta} + \frac{f(V(t))}{\tau_\theta}. \tag{58}$$

The last term (that is as a function of V) has the following expression in Fourier terms:

$$\frac{f(V(t))}{\tau_\theta} = \frac{a + e^{b(V(t)-c)}}{\tau_\theta} = \frac{\hat{a}_0}{2} + \sum_{k=1}^{n} \hat{a}_k \cos\left(\frac{2\pi k t}{T}\right) + \hat{b}_k \sin\left(\frac{2\pi k t}{T}\right). \tag{59}$$

If we consider the coefficients of (59) considering the particular solution of $V_p(t)$, then these coefficients are:

$$\begin{aligned}
\hat{a}_0 &= \frac{1}{T}\int_0^T \frac{f(V_p(t))}{\tau_\theta} dt = \frac{1}{T}\int_0^T \frac{a + e^{b(V_p(t)-c)}}{\tau_\theta} dt \\
\hat{a}_k &= \frac{1}{T}\int_0^T \frac{f(V_p(t))}{\tau_\theta} \cos\left(\frac{2\pi k t}{T}\right) dt = \frac{1}{T}\int_0^T \frac{a + e^{b(V_p(t)-c)}}{\tau_\theta} \cos\left(\frac{2\pi k t}{T}\right) dt \\
\hat{b}_k &= \frac{1}{L}\int_0^T \frac{f(V_p(t))}{\tau_\theta} \sin\left(\frac{2\pi k t}{T}\right) dt = \frac{1}{T}\int_0^T \frac{a + e^{b(V_p(t)-c)}}{\tau_\theta} \sin\left(\frac{2\pi k t}{T}\right) dt
\end{aligned} \tag{60}$$

.

We will try to find a particular solution for $\theta(t)$ in Fourier terms. For instance we propose:

$$\theta_p(t) = g(t) = \frac{\tilde{a}_0}{2} + \sum_{k=1}^{n} \tilde{a}_k \cos\left(\frac{2\pi k t}{T}\right) + \tilde{b}_k \sin\left(\frac{2\pi k t}{T}\right). \tag{61}$$

Thus, to find the solution for $\theta(t)$, we only need to find the coefficients $\tilde{a}_0$, $\tilde{a}_k$ and $\tilde{b}_k$ of the previous expression. These coefficients will be expressed in terms of $\hat{a}_0$,$\hat{a}_k$ and $\hat{b}_k$ in (59).

So differentiating the proposed particular solution in (61) and equating to (58) considering (59) we have:

$$-\frac{2\pi}{T}\sum_{k=1}^{n} k\tilde{a}_k \sin\left(\frac{2\pi}{T}kt\right) + \frac{2\pi}{T}\sum_{k=1}^{n} k\tilde{b}_k \cos\left(\frac{2\pi}{T}kt\right)$$

$$= -\frac{\tilde{a}_0}{2\tau_\theta} - \sum_{k=1}^{n}\left(\frac{\tilde{a}_k}{\tau_\theta}\cos\left(\frac{2\pi k t}{T}\right) + \frac{\tilde{b}_k}{\tau_\theta}\sin\left(\frac{2\pi k t}{T}\right)\right) + \frac{\hat{a}_0}{2} + \sum_{k=1}^{n}\hat{a}_k \cos\left(\frac{2\pi k t}{T}\right) + \hat{b}_k \sin\left(\frac{2\pi k t}{T}\right).$$

Reagruping terms for each respective coefficients from both sides of the previous expression, we arrive to the following set of equations:

$$
\begin{aligned}
\tilde{a}_0 &= \hat{a}_0\tau_\theta \\
-\frac{2\pi k}{T}\tilde{a}_k &= \hat{b}_k - \frac{\tilde{b}_k}{\tau_\theta} \\
\frac{2\pi k}{T}\tilde{b}_k &= \hat{a}_k - \frac{\tilde{a}_k}{\tau_\theta}.
\end{aligned} \tag{62}
$$

we need to solve the last two equations to express $\tilde{a}_k$ and $\tilde{b}_k$ in terms of $\hat{a}_k$ and $\hat{b}_k$. These expressions once solved are expressed as below:

$$
\begin{aligned}
\tilde{a}_0 &= \hat{a}_0\tau_\theta \\
\tilde{a}_k &= \left(\frac{T^2}{T^2 + 4\pi^2k^2\tau_\theta^2}\right)\left(\hat{a}_k\tau_\theta - \frac{2\pi k}{T}\hat{b}_k\tau_\theta^2\right) \\
\tilde{b}_k &= \left(\frac{T^2}{T^2 + 4\pi^2k^2\tau_\theta^2}\right)\left(\hat{b}_k\tau_\theta + \frac{2\pi k}{T}\hat{a}_k\tau_\theta^2\right).
\end{aligned} \tag{63}
$$

As it can be seen, the coefficients (63) are in terms of (60), where the latter are integrals for a full period $T = \frac{1}{\omega}$. Also note that these coefficients are in function of the particular solution of $V_p(t)$.

Then, the particular solution $\theta_p(t)$ is:

$$
\begin{aligned}
\theta_p(t) = \frac{\hat{a}_0\tau_\theta}{2} &+ \sum_{k=1}^{n}\left(\frac{T^2}{T^2 + 4\pi^2k^2\tau_\theta^2}\right)\left(\hat{a}_k\tau_\theta - \frac{2\pi k}{T}\hat{b}_k\tau_\theta^2\right)\cos\left(\frac{2\pi kt}{T}\right) \\
&+ \left(\frac{T^2}{T^2 + 4\pi^2k^2\tau_\theta^2}\right)\left(\hat{b}_k\tau_\theta + \frac{2\pi k}{T}\hat{a}_k\tau_\theta^2\right)\sin\left(\frac{2\pi kt}{T}\right).
\end{aligned} \tag{64}
$$

In Figure 1 and 2 we show the particular solutions V_p andθ_p as functions of time for the following values: $V_r = 0.10$,$a = 0.08$,$b = 4.9$,$c = 0.53$,$\tau = 2$, $A = 0.3$ and $\omega = 0.02$. The coefficients (60), which appear in formula (64), are computed numerically using the trapezoidal method for integrals.
In Figure 3 we show how the particular solution (V_p, θ_p) looks on the V-θ plane with the same parameters used before.

The programming code of the integrator for this time dependent current input was written in matlab and can be found on the appendix.

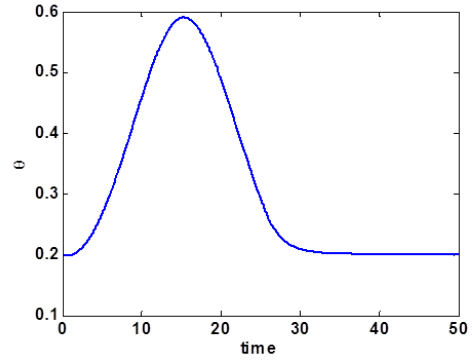

FIG. 1. θ vs. time

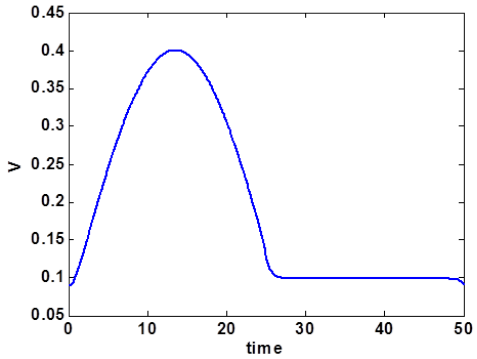

FIG. 2. V vs *time*

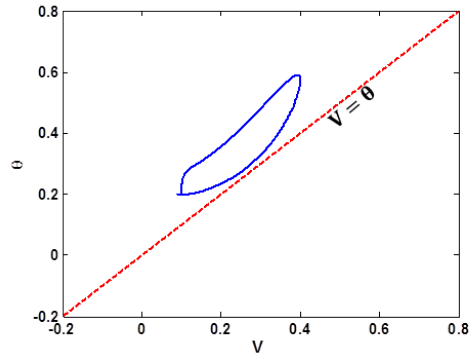

FIG. 3. θ vs. V

2.4. Qualitative approach of the differential equation of the $V-\theta$ system with Constant current input.

We focus in this section on the qualitative study of our non linear system for

a constant current input. By now we discard the non-smooth nature of it, that is the resetting conditions (25):

$$\begin{aligned} \frac{dV}{dt} &= -V + V_r + I \\ \frac{d\theta}{dt} &= -\frac{\theta - f(V)}{\tau_\theta} \end{aligned} \tag{65}$$

with

$$f(V) = a + exp(b(V - c))$$

.

We start by finding *fixed points* (V^*, θ^*) of the system (65). So we need to solve the equations:

$$\begin{aligned} 0 &= -V + V_r + I \\ 0 &= -\theta + f(V). \end{aligned}$$

This implies that our fixed points are:

$$\begin{aligned} V^* &= V_r + I \\ \theta^* &= f(V^*) \end{aligned} \tag{66}$$

The stability of the fixed point can be studied with the linearization of (65) at (V^*, θ^*). That is, we compute the Jacobian matrix for (65), evaluate at the fixed point (66), and compute the respective eigenvalues:

$$D\chi(V^*, \theta^*) = \begin{pmatrix} -1 & 0 \\ \frac{f'(V_r+I)}{\tau_\theta} & -\frac{1}{\tau_\theta} \end{pmatrix}$$

$$\lambda_1 = -1 \quad \lambda_2 = -\frac{1}{\tau_\theta}. \tag{67}$$

We compute the eigen-vectors for the respective eigenvalues. In the case for λ_1:

$$\begin{pmatrix} -1 & 0 \\ \frac{f'(V_r+I)}{\tau_\theta} & -\frac{1}{\tau_\theta} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = -1 \begin{pmatrix} a \\ b \end{pmatrix},$$

where we solve the system:

$$\begin{aligned} -a &= -a \\ a\frac{f'(V_r + I)}{\tau_\theta} - \frac{b}{\tau_\theta} &= -b \end{aligned}$$

to obtain the eigenvector.

$$u_1 = \begin{pmatrix} 1 \\ \frac{f'(V_r+I)}{1-\tau_\theta} \end{pmatrix}$$

In the case for λ_2 the associated eigenvector is:

$$u_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

One important thing is that if all the eigenvalues are real and $\lambda_1, \lambda_2 < 0$ then we have an *attracting node*. On the contrary, way if $\lambda_1, \lambda_2 > 0$ our fixed point will be a repelling node. In our case we have $\lambda_1 < 0$ and λ_2 depends on the sign of the parameter τ_θ. Since $\tau_\theta > 0$ we are considering the case of an attracting node.

Is important that for the $V - \theta$ model (65), the parameters are $a, b, c, \tau_\theta, V_r > 0$.

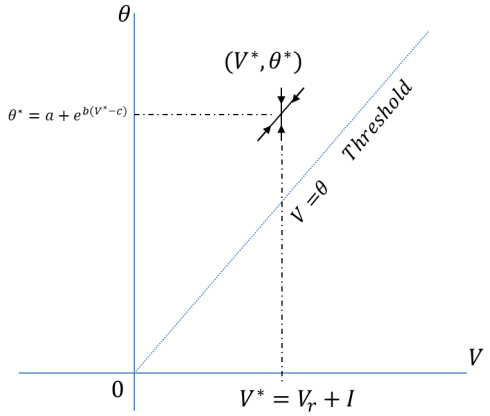

FIG. 4. Qualitative Figure of the $V - \theta$ model considering $a, b, c, \tau_\theta, V_r > 0$

Since $\lambda_1 > \lambda_2$ our solutions on the phase portrait will contract faster along the direction given by the eigenvector u_1 than u_2.

Finally if we take the solution computed previously for $V(t)$ from (27) and if we consider the limit for $t \to \infty$ we have:

$$\lim_{t\to\infty} V(t) = \lim_{t\to\infty} e^{-t}(V_0 - V_r - I) + V_r + I = V_r + I = V^*.$$

The same can be done for $\theta(t)$ from (31). However l'Hopital rule for the second term should be applied in order to evaluate the limit for $t \to \infty$:

$$\lim_{t\to\infty}\theta(t)=\theta_0 e^{\frac{-t}{\tau_\theta}}+\int_0^t \frac{e^{\frac{s-t}{\tau_\theta}}}{\tau_\theta}\left(a+e^{b(V(s)-c)}\right)ds=\lim_{t\to\infty}\frac{\theta_0}{e^{\frac{t}{\tau_\theta}}}+\frac{\int_0^t \frac{e^{\frac{s}{\tau_\theta}}}{\tau_\theta}\left(a+e^{b(V(s)-c)}\right)ds}{e^{\frac{t}{\tau_\theta}}}$$

$$\xrightarrow{l'hopital}\lim_{t\to\infty}\frac{\frac{e^{\frac{t}{\tau_\theta}}}{\tau_\theta}(a+e^{b(V(t)-c)})}{\frac{1}{\tau_\theta}e^{\frac{t}{\tau_\theta}}}=a+e^{b(V_r+I-c)}=f(V_r+I)=\theta^*.$$

Therefore (V^*,θ^*) is a fix point.

In Figure 4 we present a qualitative figure of the system.

Finally we want to state that a qualitative analysis with time dependet current input $I(t)$ will not be performed in this work. Where we are going to focus on the case with a constant current input I. However an analysis of $I(t)$ should not be discarded in the future (numerically or analytically), as we are confident that an extension of our final work is possible considering a time dependent $I(t)$.

3. Dynamics for the V-θ system with Constant Current Input.

The qualitative approach should be complemented by the analysis of results from numerical simulations. These numerical simulations are performed for the system considering a constant current input I. For the first section we discarded the post-spike resetting conditions in order to have a total figure of the $V-\theta$ phase plane. The numerical integration was coded using Matlab, and can be found in the appendix section.

3.1. A numerical integration for the differential equation for the V-θ system with Constant Current Input.

The qualitative results obtained here helped us to characterize a domain $\Omega \in \mathbb{R}^2$ where if we take initial conditions $(V_0, \theta_0) \in \Omega$ then the respective solution $\varphi(t, V_0, \theta_0)$ will approach the line $V = \theta$, spike will occur, and the post-spike resetting conditions (25) will be applied.

Different sets of initial conditions were considered for computing the numerical solutions. The system is integrated using the following values for the parameters: $a = 0.08, b = 4.9, c = 0.53, \tau_\theta = 2, V_r = 0.1$ and I taking a values for $I \in [0.1, 0.3]$.

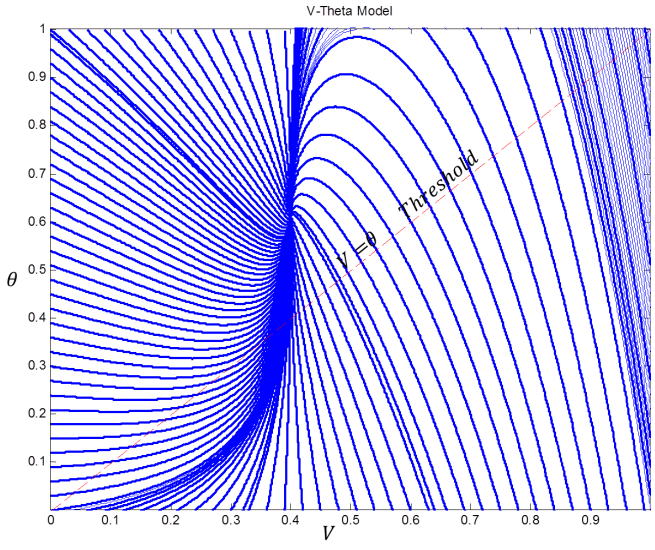

FIG. 5. Trajectories on the $V = \theta$ plane, with values $a = 0.08$,$b = 4.9$,$c = 0.53$,$V_r = 0.10$,$I = 0.30$,$\tau_\theta = 2$

In Figure 6 it can be seen that trajectories of the systems approach the node $(V^*, \theta^*) = (0.4, 0.609)$ intersecting the line $V = \theta$. Moreover there is a set of trajectories for $V < V^*$ that never intersect the $V = \theta$ line.

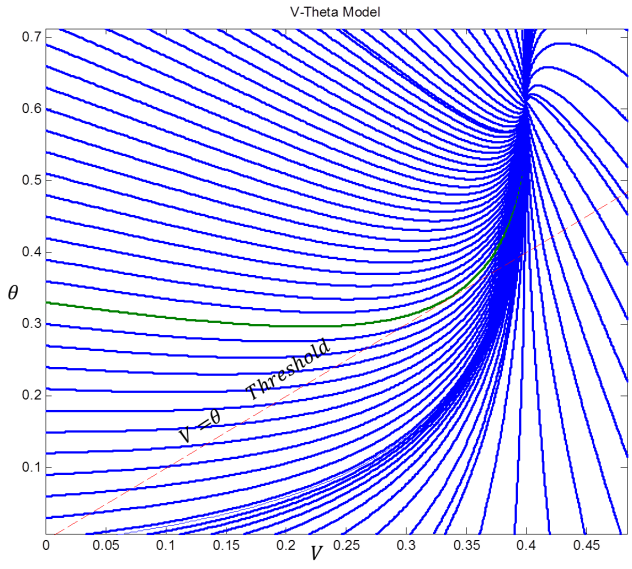

FIG. 6. Zoom applied to trajectory tangent (green line) to $V = \theta$

In Figure 6 we show a zoom of trajectories near the threshold line $V = \theta$.

Again in Figure 6 we see that some set of solutions intersects the line transversely, while other trajectories approach directly to the fix point (V^*, θ^*). Also it appears to be a unique solution $\tilde{\varphi}$ which is tangent to a point $(\tilde{V}, \tilde{\theta})$ in $V = \theta$ and finally reaches the fix point (V^*, θ^*) without intersecting again the $V = \theta$ line.

In the next section we characterize our $V - \theta$ phase plane into different regions, where the solution $\tilde{\varphi}$ will define a boundary for the set of $(V_0, \theta_0) \in \Omega$ which leads to define trajectories that will intersect the $V = \theta$ at some t.

3.2. The switching manifold and the $V-\theta$ plane.

Let us define $\Sigma_s = \{z, h(z) = 0\}$ which is the line in $\mathbb{R}^2$ that we call the switching manifold. We have that our phase plane is separated in two regions: $\Sigma^- = \{z, h(z) < 0\}$ and $\Sigma^+ = \{z, h(z) > 0\}$. All initial conditions $(V_0, \theta_0) \in \Sigma^-$ generate an immediate spike for $t = 0$ because $V > \theta$ and are reset to the point:

$$(\theta_0 + \Delta_\theta, V_{reset}) \in \Sigma_r = \{(V, \theta), V = V_{reset}\}, \tag{68}$$

for the other initial conditions $(V_0, \theta_0) \in \Sigma^+$ the solutions evolve following the differential equations (65) and approach the node (V^*, θ^*) or Σ_s at some time t.

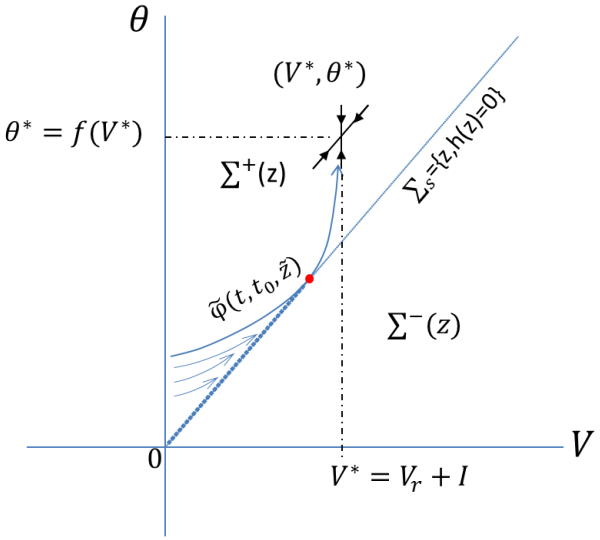

FIG. 7. There is a unique solution tangent to Σ

We know that there is a solution in Σ^+ which is tangent to the switching line Σ_s. We have that $h(z) = V - \theta$ so $\nabla h = [-1, 1]$, denoting $X(z) = [\dot{V}, \dot{\theta}]^T$, then we get:

$$X(z) \cdot \nabla h = -\dot{V} + \dot{\theta}$$

where this expression should be equal to 0 for finding the tangent point $(\tilde{V}, \tilde{\theta})$.

So, using (65) we have:

$$-\frac{\theta}{\tau_\theta} + \frac{a + e^{(b(V-c))}}{\tau_\theta} = -V + V_r + I.$$

we also know that in the tangency we have that $\theta = V$ so:

$$-\frac{\theta}{\tau_\theta} + \frac{a + e^{(b(\theta-c))}}{\tau_\theta} = -\theta + V_r + I.$$

which is equivalent to:

$$-\theta + V_r + I + \frac{\theta}{\tau_\theta} - \frac{a + e^{(b(\theta - c))}}{\theta} = 0.$$

This equation is solved numerically, it can be seen that there is only one value for $\tilde{\theta}$ that satisfies this equation, confirming that *there is only one solution* $\tilde{\varphi}(t, \tilde{\theta}, \tilde{\theta})$ *tangent to* $V = \theta$, and all the solutions that are below this $\tilde{\varphi}$ will intersect the line $V = \theta$ and will generate at least one spike.

3.3. Characterization of the Spiking Process.

Once stated that all the solutions that are below $\tilde{\varphi}$ define an spike, we can proceed to characterize these solutions based on the number of spikes they will produce.
Consider the solution $\tilde{\varphi}_1(t) = \varphi(t, \tilde{z}_1)$ such that $\tilde{\varphi}_1 = \tilde{z}_1$, where $\tilde{z}_1 = \varphi(t, \tilde{\theta}, \tilde{\theta}) \in \Sigma_s$. If we integrate backwards in time until $\tilde{\varphi}_1(t_1) \in \Sigma_r$, then this intersection with Σ_r can be defined by $(V_1, \theta_1) \in \Sigma_r$. We define this segment of $\tilde{\varphi}_1$ from $t_1 \leq t \leq 0$ as $S_1 = \{\varphi(t, \tilde{z}_1), t_1 \leq t \leq 0\}$ see Figure.
From the previous point, then $\tilde{\theta}_2 = \theta_1 - \Delta_\theta$, and which define $\tilde{z}_2 = (\tilde{\theta}_2, \tilde{\theta}_2) \in \Sigma_s$. This $\tilde{z}_2$ will define $\tilde{\varphi}_2 = \varphi(t, \tilde{z}_2)$ such that $\varphi(0, \tilde{z}_2) = \tilde{z}_2$, and again we integrate backwards in time until $\tilde{\varphi}_2(t_2) \in \Sigma_r$, so we define $S_2 = \{\varphi(t, \tilde{z}_2), t_2 \leq t \leq 0\}$.

We can describe a region Ω_1 as the one defined between S_1 and S_2. In conclusion any $\varphi(t, z_0) \in \Omega_1$ with $z_0 \in \Omega_1$ will lead to solutions that will intersect Σ_s once.
Similarly a region Ω_2 can be defined a segment S_3, where Ω_2 is defined between S_2 and S_3, and where $\varphi(t, z_0) \in \Omega_2$ with $z_0 \in \Omega_2$ will lead to solutions that will intersect Σ_s two times before reaching the fix point (V^*, θ^*).

Different Ω_i regions can be obtained as long as $\tilde{z}_i > 0$ can be defined. In Figure 8 we show a phase portrait where the regions are already defined. The values taken are $a = 0.08$, $b = 4.9$, $c = 0.53$, $V_r = 0.10$, $I = 0.30$, $\tau_\theta = 2$ and $\Delta_\theta = 0.1$.

In Figure 9 we show an example considering an initial condition $z_0 \in \Omega_2$ which generates two spikes before approaching the node.

On Fig. 10 we show the two components of the solution $\varphi(t, z_0) = (V(t), \theta(t))$ as function of time. It is evident the behaviour of the θ threshold variable and how a spike is done everytime the *voltage* reaches the θ threshold and finally converging to the fix point (V^*, θ).

3.4. Future studies to the V-θ with periodic time dependent current input. As we previously said, an analysis with $I(t)$ was not performed in this work, however with the previous Fourier solutions found and the previous numerical solver programmed with Fourier series it can be possible to perform a study for the V-θ system to find a set of A and ω values for the periodic half wave rectified sinusoid

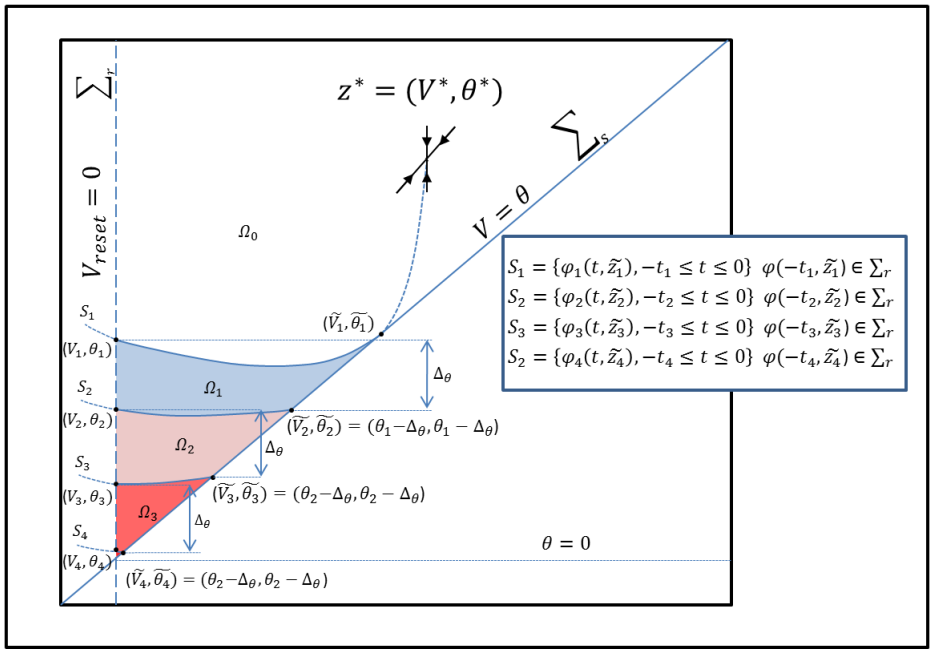

FIG. 8. Phase Phortrait characterized by different Ω regions.

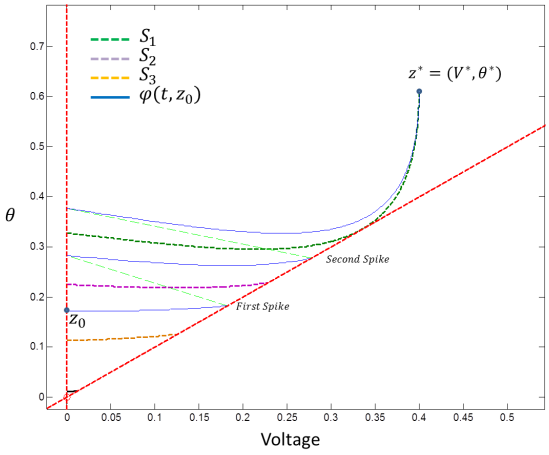

FIG. 9. Initial condition $z_0 \in \Omega_2$ which creates two spike.

$I(t)$ that will lead to the periodic orbit from V_p and θ_p to cross the bisecting line $V = \theta$, see Fig:11.

This can be a potential goal to deepen the study in [4], where set of values for A (amplitude), ω (frequency) are considered in order to classify which parameters

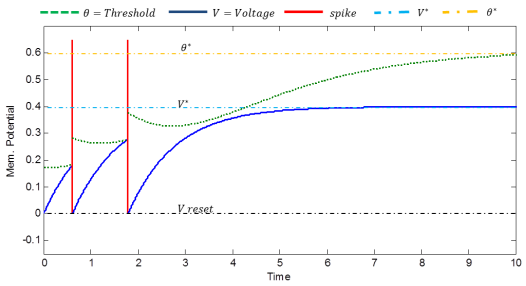

FIG. 10. *Time* vs *Mem.Potential* chart from previous example.

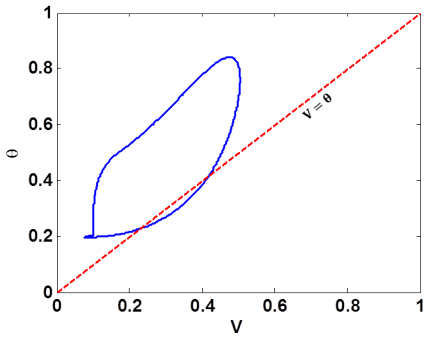

FIG. 11. Periodic current $I(t)$, with $A = 0.04$ and $\omega = 0.04$ cross $v = \theta$ line.

produce one spike or none.

The next goal is to implement a simpler neuron model with a dynamic threshold that get rids of the non-linearity of the V-θ model. This new proposed model will be based on a modification of the resonate and fire model and will have a dynamic threshold and resetting conditions.

Chapter 3
Modified Resonate and Fire model with Dynamical Threshold

1. The model

The modified resonate and fire model ($MR\&F$) is given by the following set of equations:

$$\begin{aligned} \dot{x} &= bx - \omega y + I \\ \dot{y} &= \omega x + by \end{aligned} \tag{69}$$

where I is a constant current input of the system. The classical resonate and fire model [3] considers a threshold for a given $y = 1$ value, and was presented previously in chapter 1. The difference between this model and the $V - \theta$ model introduced in chapter 2 is that the first one is a resonator and the second one an integrator. In this section we propose a system based on the Resonate and Fire model of Izhikevich [1],[3], which in some way could simplify the analytic treatment that is not possible for the model of the $V - \theta$ system with periodic input. We will keep the reset (non-smooth) conditions of the $V - \theta$ and the differential equations of the resonate and fire model. Although both models include the dynamic threshold the subtreshold dynamics will be different because the critical point is now a focus instead of a node. This will create a richer dynamics that we will explore in this section. Consider the switching manifold $\Sigma_1 = \{z, h(z) = 0\}$, where $h(z) = h(x, y) = x - y$. The resetting conditions of the model (69) are imposed every time a spike is generated that is when $x(t_f) = y(t_f)$. Where t_f denotes the time of the spiking moment. These conditions are the following:

$$\begin{aligned} x(t_f^+) &= V_{res} \\ y(t_f^+) &= y(t_f) + \Delta_y \end{aligned} \tag{70}$$

where V_{res} is the reset value of the neuron after a spike. The line $x = V_{res}$ is denoted as Σ_0. In this section we consider $V_{res} < x_e^*$ where $z_e^* = (x_e^*, y_e^*)$ denotes the equilibrium point of the system (69). More specifically we will use the value of $V_{res} \in (-0.13, 0.01)$ for $b = -1$ and $\omega = 10$ as the original resonate and fire model

in [1] and the constant current will be $I = 1$. In the case of Δ_y we will explain the different values it can take in the following sections.

1.1. Qualitative properties of the differential equation.

The linear system (69) is of the type:

$$\dot{z} = Az + f, \tag{71}$$

where A is a constant matrix and f the constant column vector:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} b & -\omega \\ \omega & b \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} I \\ 0 \end{pmatrix}.$$

The equilibrium points are given by:

$$\begin{aligned} x_e^* &= -\frac{bI}{b^2+\omega^2}, \\ y_e^* &= \frac{\omega I}{b^2+\omega^2}, \end{aligned} \tag{72}$$

where the eigenvalues of the matrix A are:

$$\lambda_{1,2} = b \pm \omega i,$$

and the respective eigenvectors:

$$\vec{v_1} = \begin{pmatrix} 1 \\ -i \end{pmatrix} \quad \vec{v_2} = \begin{pmatrix} 1 \\ i \end{pmatrix}.$$

We know from the complex conjugate eigenvalues that the equilibrium point of (72) focus. As $b < 0$ it is a stable focus.

2. Qualitative properties of the model.

2.1. Different cases of the $MR\&F$. Different cases can appear in the MR&F model based on the position of Σ_0. Recall that Σ_1 describes the line $x = y$ and consider a point $p_b \in \Sigma_1$. We denote φ_{tan} a particular solution $\varphi_{tan}(t) = \varphi(t, p_b)$ for (69) such that $\varphi_{tan}(0) = p_b$, and $\varphi_{tan}(t)$ intersects once tangentially Σ_1 for $t_1 > 0$. This point of intersection is defined as $p_{tan} = (x_{tan}, y_{tan}) = \varphi_{tan}(t_1)$ where $x_{tan} = y_{tan} = \frac{I}{2\omega}$, see Fig. 2. The different cases are categorized if Σ_0 intersects with $\left\{z \in \mathbb{R}^2 | z = \varphi_{tan}(t), 0 < t < t_1\right\}$:

- **Case 1**
  Values of $V_{res} < V_{res}^*$ such that Σ_0 does not intersect the curve φ_{tan}. However there exist a solution φ_{tan} which is tangent to Σ_0 at the point $p_{res}^a \in \Sigma_0$ and

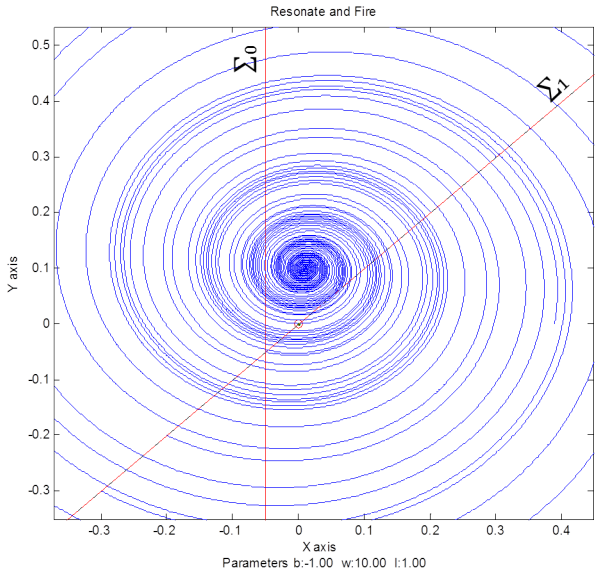

FIG. 1. Phase Phortrait, Σ_0 and Σ_1

which intersects Σ_1 at a point defined as p_c. See Fig.2.

- **Case 2**
  When $V_{res} = V^*_{res}$, Σ_0 intersects tangentially the curve φ_{tan}. The intersection point is referred as $p^*_{res} = (V^*_{res}, y^*_{res})$. For the point p^*_{res}, then $y = y^*_{res}$ can be computed using that $\dot{x} = 0$ and $x = V_{res}$. Then $y^*_{res} = \frac{I+bV^*_{res}}{\omega}$. See Fig. 3.

- **Case 3**
  The values for $V_{res} > V^*$ such that Σ_0 intersects φ_{tan} in two different points. The intersections are $p^1_{tan} = (V_{res}, y^1_{tan})$ and $p^2_{tan} = (V_{res}, y^2_{tan})$ where $y^1_{tan} < y^2_{tan}$. See Fig. 4.

For all the cases, we define the points $p_{res} \in \Sigma_0 \cap \Sigma_1$ and $\bar{p}_{res} \in \Sigma_0$ such that the solution $\varphi(t, \bar{p}_{res})$ of (69) for $t > 0$ will intersect for first time the line Σ_0 at the point p_{res}.

In Figure 5 we show how the points $p_{res}, p^1_{tan}, p^2_{tan}$ vary based on the parameter V_{res}.

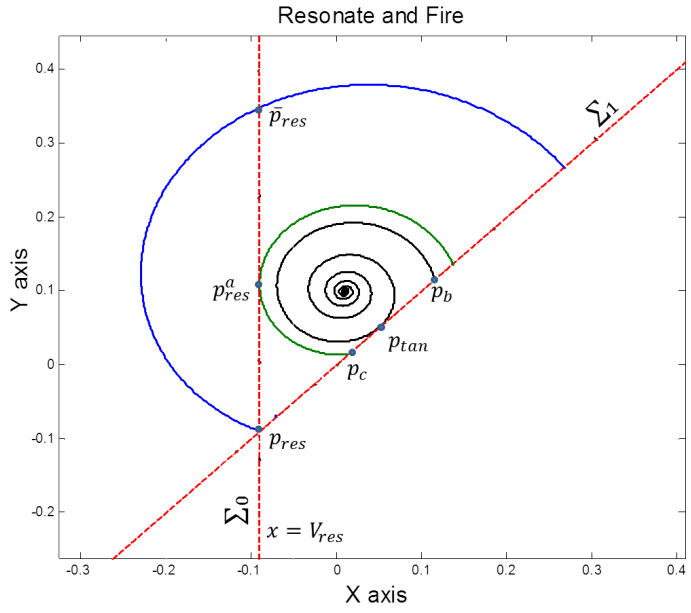

FIG. 2. Case 1

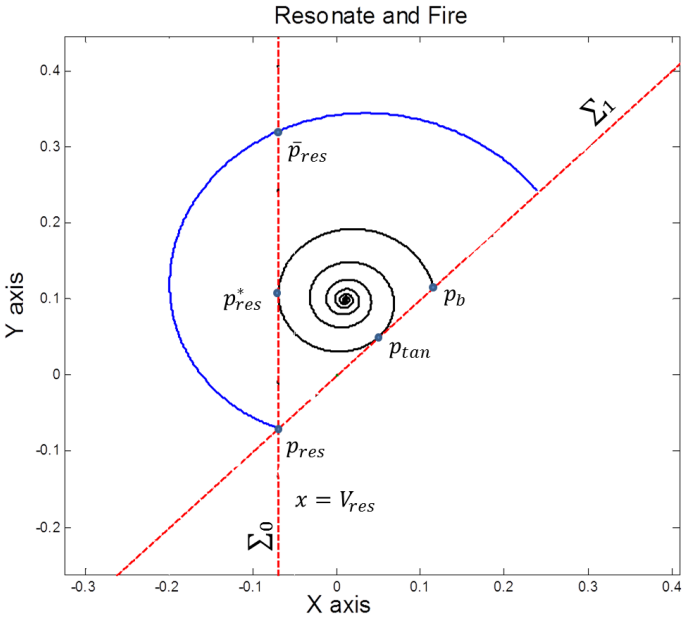

FIG. 3. Case 2

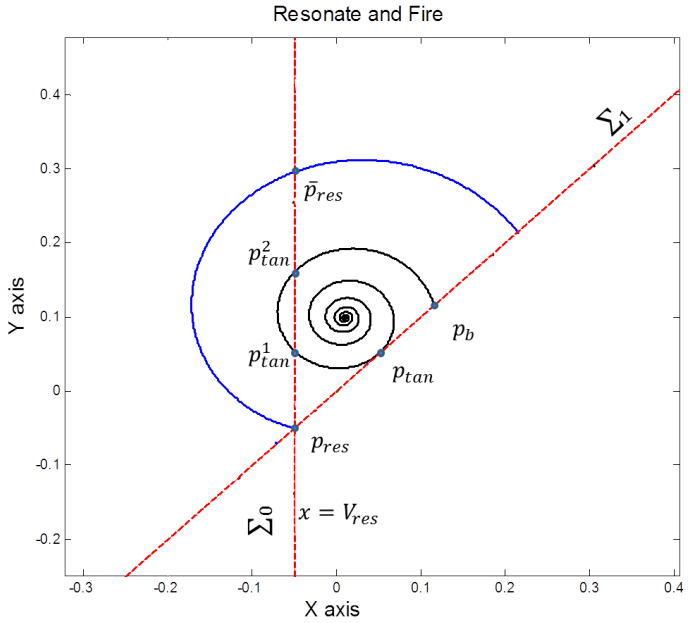

FIG. 4. Case 3

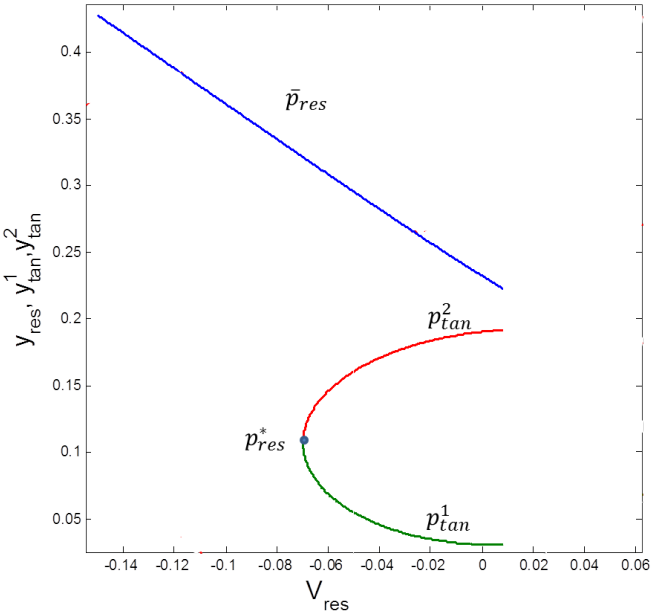

FIG. 5. Position of y_{res}, y^1_{tan}, y^2_{tan}

2.2. Solution in polar coordinates of the Vector Field.

If we consider the system (69) discarding the column vector f, our equilibrium points are $x_e^* = 0$ and $y_e^* = 0$. And applying the polar change of coordinates:

$$x = r\cos\theta, \quad y = r\sin\theta, \quad r = \sqrt{x^2+y^2}, \quad \tan\theta = \frac{x}{y},$$

we get to the system:

$$\begin{aligned} \dot{r} &= br \\ \dot{\theta} &= \omega \end{aligned} \tag{73}$$

where the solutions of such system of equations are given by:

$$r(t) = r_0 e^{bt}, \quad \theta(t) = \omega t + \theta_0,$$

for the following initial conditions: $\theta(0) = \theta_0$ and $r(0) = r_0$.

As we want a solution for the system (69), then we can use the previous solutions and apply another change of coordinates to position them in the equilibrium point (72). We get the solutions for (69):

$$\begin{aligned} x(t) = x_e^* + r\cos\theta &= -\frac{bI}{b^2+\omega^2} + r_0 e^{bt}\cos(\theta_0+\omega t) \\ y(t) = y_e^* + r\sin\theta &= -\frac{\omega I}{b^2+\omega^2} + r_0 e^{bt}\sin(\theta_0+\omega t) \end{aligned} \tag{74}$$

where r_0 and θ_0 are the initial conditions. However we can express θ_0 and r_0 in terms of x_0 and y_0. For doing that we consider that:

$$\begin{aligned} x(0) = x_0 = -\frac{bI}{b^2+\omega^2} + r_0\cos(\theta_0) \to r_0\cos(\theta_0) = x_0 + \frac{bI}{b^2+\omega^2}, \\ y(0) = y_0 = \frac{\omega I}{b^2+\omega^2} + r_0\sin(\theta_0) \to r_0\sin(\theta_0) = y_0 - \frac{\omega I}{b^2+\omega^2}, \end{aligned} \tag{75}$$

solving the previous equations for θ_0 and r_0 we get that:

$$\begin{aligned} \tan(\theta_0) &= \left(\frac{y_0(b^2+\omega^2) - \omega I}{x_0(b^2+\omega^2) + bI}\right) \\ r_0 &= \sqrt{\left(x_0 + \frac{bI}{b^2+\omega^2}\right)^2 + \left(y_0 - \frac{\omega I}{b^2+\omega^2}\right)^2}. \end{aligned} \tag{76}$$

2.3. The Return Map P. We want to find and characterize the periodic orbits of our model for a given Δ_y. For doing that first we define the following mapping:

$$\begin{aligned} P_1 : \Sigma_1 &\to \Sigma_0 \\ (x, y = x) &\mapsto (V_{res}, y + \Delta_y) \end{aligned} \tag{77}$$

where all the values for $z = (x, y) \in \Sigma_1$ satisfy that $y = x$, that is when we reach the threshold we apply the resetting conditions for arriving to Σ_0.

Now consider a second mapping:

$$\begin{aligned} P_2 : \Sigma_0 &\to \Sigma_1 \\ (x_0 = V_{res}, y_0) &\to (x_f = y_f, y_f = P_2(y_0)) \end{aligned} \tag{78}$$

where $z_0 = (x_0, y_0) \in \Sigma_0$ and $z_f = (x_f, y_f) \in \Sigma_1$ is given by $z_f = \varphi(t_f, z_0)$, where φ is the flow of system (71), so any initial condition $z_0 \in \Sigma_0$ after a given time t_f reaches Σ_1 and generates a spike.

Being more specific the second mapping is defined by the formulas:

$$\begin{aligned} x_f = \varphi^x_{t_f}(t_f, V_{res}) &= -\frac{bI}{b^2 + \omega^2} + r_0 e^{bt_f} \cos(\omega t_f + \theta_0) \\ y_f = \varphi^y_{t_f}(t_f, y_0) &= \frac{\omega I}{b^2 + \omega^2} + r_0 e^{bt_f} \sin(\omega t_f + \theta_0) \end{aligned} \tag{79}$$

where these solutions are from (74), and the time t_f is found setting $x(t_f) = y(t_f)$.

Based on the previous mappings we can define the overall map P

$$\begin{aligned} P : \Sigma_0 &\to \Sigma_0 \\ (V_{res}, y_0) &\to (V_{res}, P(y_0)) \end{aligned}$$

$$P(y_0) = (P_1 \circ P_2)(y_0) = P_1(x_f, y_f) = x_f + \Delta_y \tag{80}$$

The map P is defined for the 3 cases discussed in section 2.1, where the only restriction is its domain D_P.

For the case 1 and 2 the map P is defined when:

$$y_0 \in [V_{res}, \bar{y}_{res}] = D_P \tag{81}$$

For the case 3, the map P is defined when:

$$y_0 \in [V_{res}, y^1_{tan}] \cup [y^2_{tan}, \bar{y}_{res}] = D_P \tag{82}$$

The parameter Δ_y needs to be restricted to an interval of values in order that any *post-spike* condition never exceed the $\max(D_P) = \bar{y}_{res}$. For the *case 1* this is defined by considering the solution $\varphi(t)$ which intersects tangentially at the point $p^a_{res} \in \Sigma_0$ and then arrives to the point $p_c = (x_c, y_c) \in \Sigma_1$ (see fig. 3), then the maximum Δ^{max}_y is defined by:

$$\Delta_y^{max} = \bar{y}_{res} - y_c. \tag{83}$$

For *case 2* and *case 3* Δ_y^{max} can be defined by (see fig. 4):

$$\Delta_y^{max} = \bar{y}_{res} - y_{tan}. \tag{84}$$

From now on we consider Δ_y which can take only values in the interval $0 < \Delta_y < \Delta_y^{max}$ and which guarantees that any $z_0 \in D_P \subset \Sigma_0$ when arrives Σ_1 with a t_f then the new *post-spike* conditions will not surpass $\bar{y}_{res}$. With this restriction we can say that $P(D_P) \subset D_P$.

We now proceed to compute the map P more explicitly:

$$x(t) = -\frac{bI}{b^2+\omega^2} + e^{bt}\left(r_0\cos(\theta_0)\cos(\omega t) - r_0\sin(\theta_0)\sin(\omega t)\right).$$

Now we use (75) considering $(x_0 = V_{res}, y_0)$, so we get:

$$x(t) = -\frac{bI}{b^2+\omega^2} + e^{bt}\left(\left(V_{res} + \frac{bI}{b^2+\omega^2}\right)\cos(\omega t) - \left(y_0 - \frac{\omega I}{b^2+\omega^2}\right)\sin(\omega t)\right). \tag{85}$$

We do the same for $y(t)$, first we expand it:

$$y(t) = \frac{\omega I}{b^2+\omega^2} + e^{bt}\left(r_0\sin(\theta_0)\cos(\omega t) + r_0\cos(\theta_0)\sin(\omega t)\right)$$

and use (75) considering $x_0 = V_{res}$, we get:

$$y(t) = \frac{\omega I}{b^2+\omega^2} + e^{bt}\left(\left(y_0 - \frac{\omega I}{b^2+\omega^2}\right)\cos(\omega t) + \left(V_{res} + \frac{bI}{b^2+\omega^2}\right)\sin(\omega t)\right). \tag{86}$$

The expressions (85) and (86) are solutions for any $x_0 = V_{res}$.

We want to find the time t_f for which these initial conditions reach the threshold $x = y$. Thus we set:

$$x(t_f) = y(t_f),$$

and using the expressions (85) and (86) and reordering terms the next equation is obtained:

(87)
$$N(t,y_0) = \frac{I(b+\omega)}{b^2+\omega^2} + e^{bt}\left\{\left(y_0 - V_{res} - \frac{I(b+\omega)}{b^2+\omega^2}\right)\cos(\omega t) + \left(V_{res} + y_0 + \frac{I(b-\omega)}{b^2+\omega^2}\right)\sin(\omega t)\right\} = 0.$$

Solving equation (87) we obtain the time t_f at which the threshold is reached. This time is a function of the initial conditions V_{res} and y_0, more specifically only y_0, due to the fact that we are considering always $x_0 = V_{res}$ after a spike, where y_0 is a dynamic variable every time an spike is reached. Then it can be stated that $t_f = t_f(y_0)$ as the time of a spike for a given initial condition.

Based on the previous computations we can use (86) to redefine the mapping (78) from $P_2 : \Sigma_0 \to \Sigma_1$ and express it as:

(88)
$$\begin{aligned} P_2(y_0) &= m(y_0, t_f(y_0)) \\ &= \frac{\omega I}{b^2+\omega^2} + e^{bt_f(y_0)}\left(\left(y_0 - \frac{\omega I}{b^2+\omega^2}\right)\cos(\omega t_f(y_0)) + \left(V_{res} + \frac{bI}{b^2+\omega^2}\right)\sin(\omega t_f(y_0))\right). \\ &= y(t_f(y_0)) \end{aligned}$$

The other mapping (77) is $P_1 : \Sigma_1 \to \Sigma_0$, that is the addition of Δ_y after reaching the threshold.

From these expressions, the mapping from $\Sigma_0 \to \Sigma_0$ is defined as:

$$P(y_0) = (P_1 \circ P_2)(y_0) = m(y_0, t_f(y_0)) + \Delta_y \tag{89}$$

with a $t_f(y_0)$ that verifies $N(t, y_0) = 0$.

So with the previous equations the map $P(y_0)$ can be defined for any $y_0 \in D_P \subset \Sigma_0$ allowed for its respective case.

2.4. Fixed points of the map P and their stability.

Using the definition of the mapping, a *fixed point* for map P is a point $p = (V_{res}, y_0)$ where:

$$P(y_0) = (P_1 \circ P_2)(y_0) = m(y_0, t(y_0)) + \Delta_y = y_0. \tag{90}$$

Given an initial condition (V_{res}, y_0), after t_f and applying the *post-spike* conditions will led the dynamics to the same initial conditions.

We will be able to find fixed points y^* of the map P when $y_0 \in D_P$ is on the appropriate domain that were previously defined for each case. The fixed points of the map will define a periodic orbit of the system, whether this could be an attracting

or repelling orbit, this will be possible to know with the stability information of the fixed point.

The stability of the fixed point y^* is based on the value of the derivative of the map P at the point y^*: A fixed point is *stable* when the $|P'(y^*)| < 1$, and an *unstable* fixed point happens when $|P'(y^*)| > 1$, where $P'(y)$ is obtained by differentiating (89) respect to y_0:

(91)
$$\begin{aligned} P'(y_0) = \frac{dm(y_0, t(y_0))}{dy_0} =& \frac{\partial m(y_0, t(y_0))}{\partial y_0} + \frac{\partial m(t, y_0)}{\partial t} t'(y_0) \\ =& e^{bt}\cos(\omega t) + e^{bt}t'\left(y_0 - \frac{\omega I}{b^2+\omega^2}\right)(b\cos(\omega t) - \omega\sin(\omega t)) + \\ & e^{bt}t'\left(V_{res} + \frac{bI}{b^2+\omega^2}\right)(b\sin(\omega t) + \omega\cos(\omega t)), \end{aligned}$$

where $t'(y_0)$ can be computed from (87), that is $\frac{dN(y_0,t(y_0))}{dy_0} = \frac{\partial N}{\partial y_0} + \frac{\partial N(t,y_0)}{\partial t} t'(y_0)) = 0$ and after isolating t' we arrive to:

(92)
$$t'(y_0) = -\frac{\cos(\omega t) + \sin(\omega t)}{\left(y_0 - Vres - \frac{I(b+\omega)}{b^2+\omega^2}\right)(b\cos(\omega t) - \omega\sin(\omega t)) + \left(y_0 + Vres + \frac{I(b-\omega)}{b^2+\omega^2}\right)(b\sin(\omega t) + \omega\cos(\omega t))}.$$

From (92) with a fix point y^* and $t(y^*)$ then $t'(y^*, t_f)$ is obtained so we can evaluate (91) for getting the stability of y^*.

The map P depends on two parameters V_{res} and Δ_y. In the following sections we proceed to vary these parameters in order to find the fixed points. First we fix the value for V_{res} and vary Δ_y. This is done for the different cases ($V_{res} < V^*_{res}, V_{res} = V^*_{res}, V_{res} > V^*_{res}$). After that we now fix Δ_y and vary V_{res}, where V_{res} will be varied on an interval $V_{res} \in [-0.13, V^*_{res}]$ (case 1,2), and then for another segment $V_{res} \in (V^*_{res}, x^*_e]$ (case 3), where x^*_e is the x variable of the stable focus from the $MR\&F$ model.

The results obtained for each case are explained, where each particular case shows different nature of the mappings and their fixed points. For all the computations the values are fixed for $\omega = 10$, $b = -1$ and $I = 1$, and for those values $V^*_{res} = -0.0698$.

2.4.1. **Case 1: $V_{res} < V^*_{res}$.**

In this section we study the map P defined in (89) using 3 different V_{res} from case 1 and a specific Δ_y for the respective domain D_P (81). We compute the values of the fixed points y^* with (90) for the whole range $\Delta_y \in [0, \Delta^{max}_y]$ where Δ^{max}_y is defined by (83), and finally we show the stability of these fixed points.

The computation for 3 different maps P is showed at Figure 6 with the different $V_{res} = -0.08, -0.09, -0.1$ values with the same $\Delta_y = 0.015$. It shows how taking more negative values of V_{res} define less concave maps.
Taking the case when $V_{res} = -0.09$, we calculated Δ_y^{max} and computed the maps for 3 different $\Delta_y = 0, \Delta_y < \Delta_y^{max}, \Delta_y^{max}$ as shown in Figure 7. The map intersects once the bisecting line $z_0 = y_0$. This intersection defines a fixed point y_s^*. We donet $y_s^{*(i)}$ as the fixed point when $\Delta_y = 0$ and $y_s^{*(f)}$ when $\Delta_y = \Delta_y^{max}$. As the map is continuous in D_P we can guarantee that there exists one fixed point $y_s^* \in [y^{*(i)}, y^{*(f)}] \subset D_P$ always for $\Delta_y \in [0, \Delta_y^{max}]$.

In Figure 8 the diagram shows the fixed points for $V_{res} = -0.08, -0.09, -0.1$ varying the parameter $\Delta_y \in [0, \Delta_y^{max}]$, which is a continuous curve with no bifurcations with always stable fixed points y_s^*. In Figure 9 the derivative of the fixed points is shown and the values are $|DP(y_s^*)| < 1$ for all $\Delta_y \in [0, \Delta_y^{max}]$ and $V_{res} \in [0.13, V_{res}^*)$. We can confirm at least locally that there is only one fixed points for case 1 that is always stable.

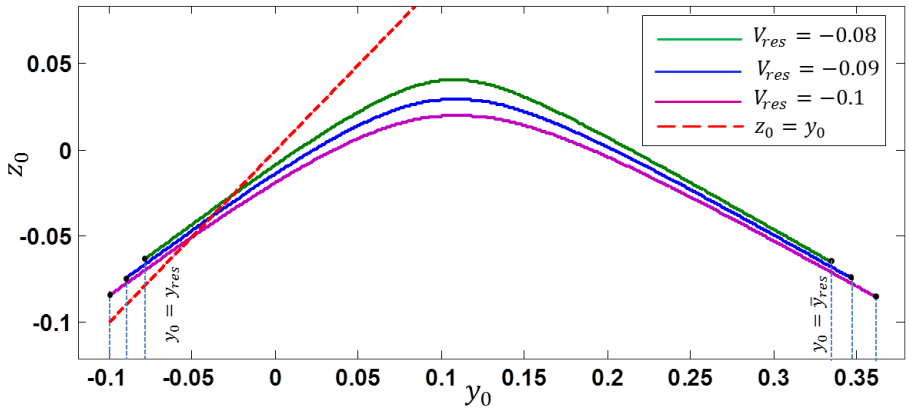

FIG. 6. **Case 1** mapping plot Σ_0 to Σ_0, with three diferent $V_{res} <= V_{res}^*$ and fixed $\Delta_y = 0.015$

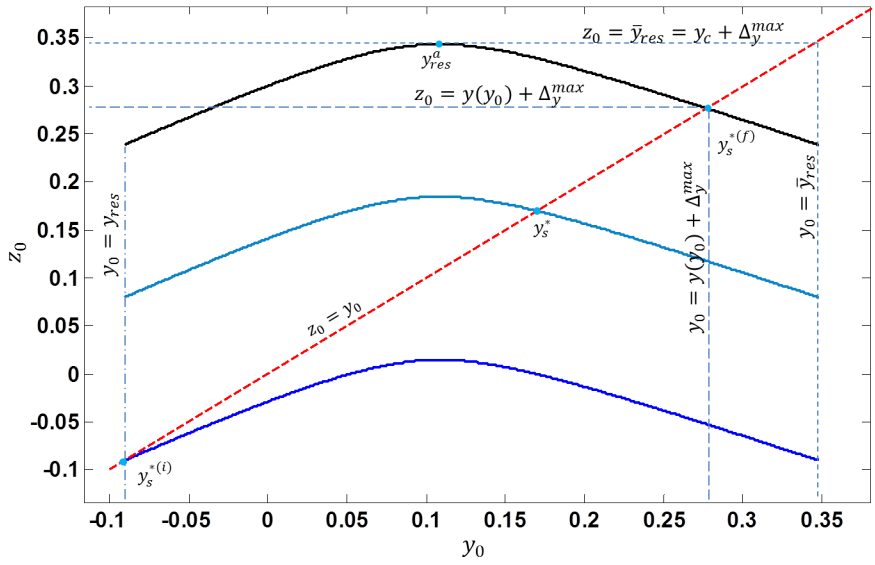

FIG. 7. **Case 1** mapping plot for $V_{res} = -0.09$, where $\Delta_y = 0, 0.17, \Delta_y^{max}$; where $\Delta_y^{max} = 0.33$

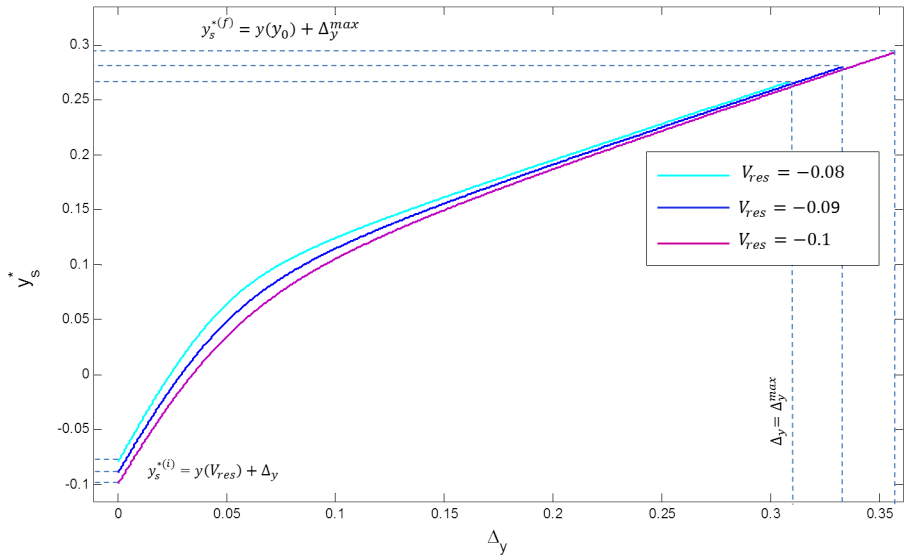

FIG. 8. **Case 1** fixed points varying $\Delta_y \in [0, \Delta_y^{max}]$

2.4.2. **Case 2:** $V_{res} = V_{res}^*$.

It happens to be pretty similar to case 1. Here we study the map P with $V_{res} = V_{res}^*$ and a specific Δ_y. We look for the fixed points for this case 2 with $\Delta_y \in [0, \Delta_y^{max}]$

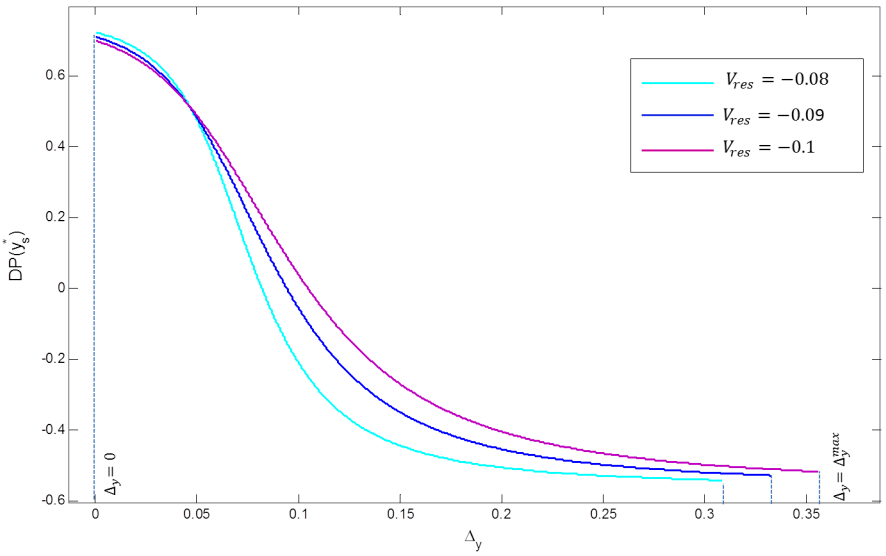

FIG. 9. **Case 1** derivative of the fixed points shows stable values with $|D'P(y_s^*)| < 1$

where Δ_y^{max} is defined by (84), and finally the stability of these fixed points.
In Figure 10 the map P is plotted. Notice that it is similar to case 1. However the concavity of the map P is more pronounced compared to Case 1. Similarly for each $\Delta_y \in [0, \Delta_y^{max}]$ there exists only a fixed point y_s^* which is *always* stable. In Figure 12 there is the diagram of the fixed point with respect to the parameter Δ_y. Figure 13 shows its derivative, that always satisfies $|DP(y_s^*)| < 1$ for $\Delta_y \in [0, \Delta_y^{max}]$.

2.4.3. **Case 3:** $V_{res} > V_{res}^*$.

The domain $D_P = [V_{res}, y_{tan}^1] \cup [y_{tan}^2, \bar{y}_{res}]$ is disconnected in two subdomains. In this section we computed the map P for three different values of V_{res} and a fixed Δ_y and explain the fixed points for both subdomains. Then we show the fixed point diagram explaining how for the first subdomain a saddle-node bifurcation takes places and for the second subdomain the fixed point changes from unstable to stable by varying Δ_y. We will see that for small values of Δ_y the fixed points are in the first subdomain whereas for bigger values of Δ_y the fixed points is in the second subdomain.

- **First subdomain** $[V_{res}, y_{tan}^1]$**:**
  The computation of the map P for case 3 for the respective domain D_P (82) is shown in Figure 14 with three different $V_{res} = -0.04, -0.05, -0.06$ values and the same $\Delta_y = 0.135$. In this figure the left hand side segment belongs to the first subdomain. In the case $V_{res} = -0.06$ the first subdomain has one stable fixed point y_s^* as there is only one intersection with the bisecting line $z_0 = y_0$. For $V_{res} = -0.05$ there are two intersections with $z_0 = y_0$ and two fixed points exists, one unstable and one stable $y_{u,s}^*$. In Figure 15 a more detailed figure

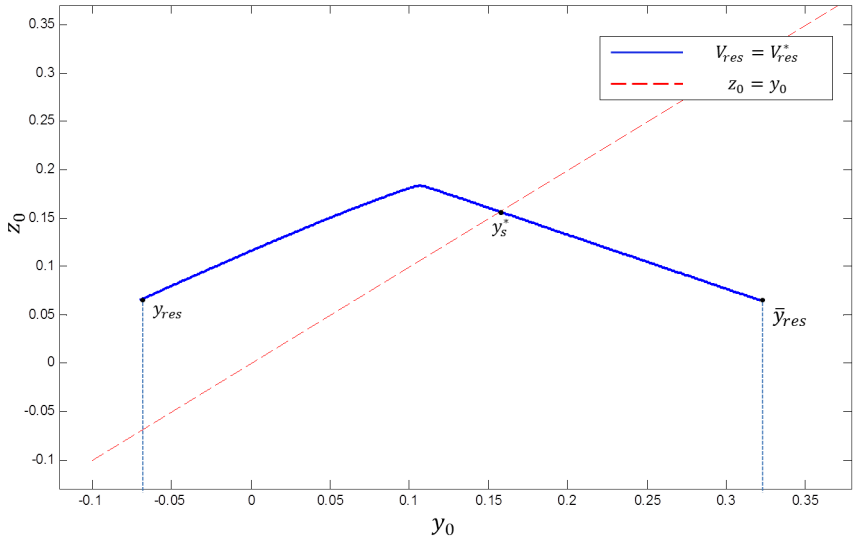

FIG. 10. **Case 2** mapping plot for $V_{res} = V_{res}^*$, and fixed $\Delta_y = 0.135$

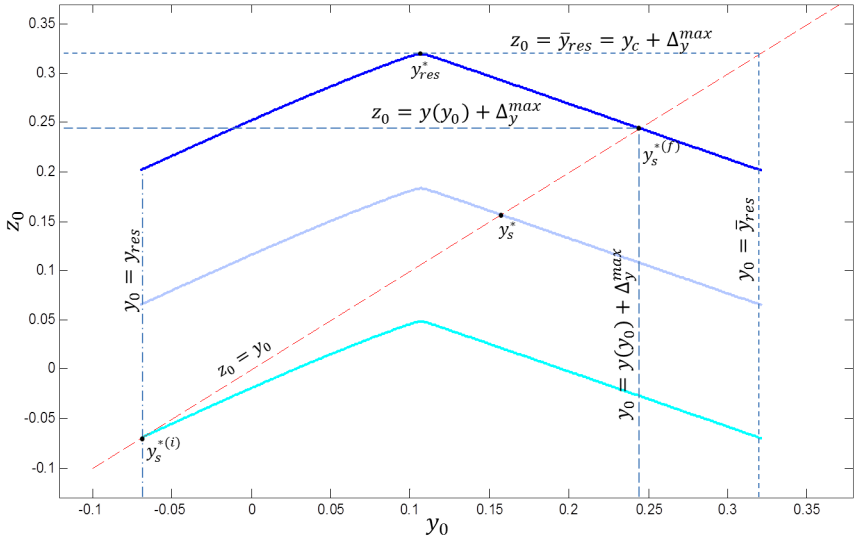

FIG. 11. **Case 2** mapping plot where $\Delta_y = 0, 0.135, \Delta_y^{max}$ where $\Delta_y^{max} = 0.2716$

for $V_{res} = -0.05$ with fixed $\Delta_y = 0.013$ shows the two fixed points y_s^* and y_u^*.

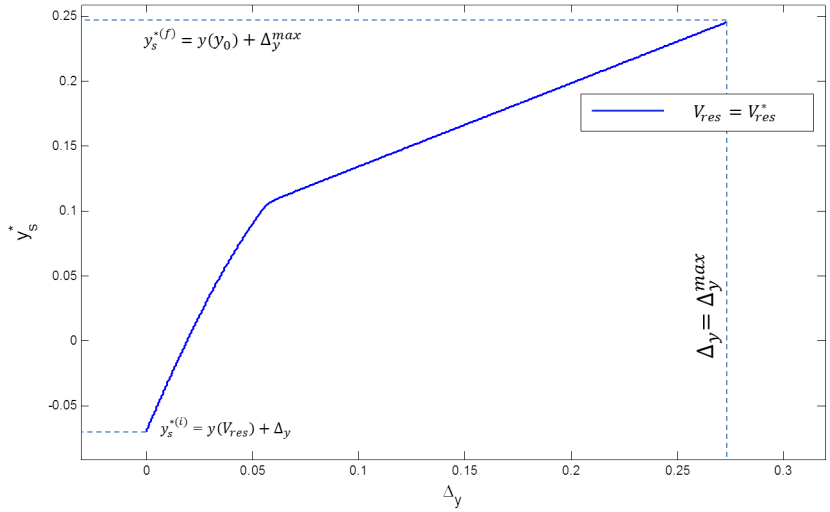

FIG. 12. **Case 2** fixed points varying $\Delta_y \in [0, \Delta_y^{max}]$

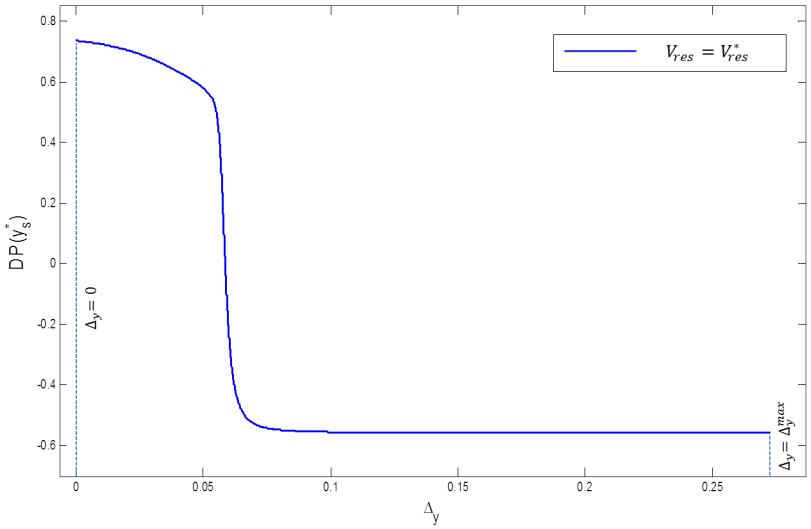

FIG. 13. **Case 2** derivative of the fixed points shows stable values with $|D'P(y_s^*)| < 1$

Figure 17 shows the fixed point diagram for three different $V_{res} = -0.04, -0.05, -0.06$ values varying $\Delta_y \in [0, y_{tan}^1 - \Delta_y]$, where the left hand side of the figure represents the fixed points of the first subdomain. A zoom of the left hand side is in Figure 18. It shows more clearly the bifurcation, which is a saddle-node bifurcation. Notice that a unique y_s^* can exist for some Δ_y values. This happens because the domain D_P is discontinous and this does not allow the y_u^*

to exist for some $\Delta_y \in [0, \Delta_y^{max}]$ values. For instance for the first subdomain $[y_{res}, y_{tan}^1]$ it can happen that just a stable y_s^* fixed point exists, or that two fixed points exist $y_{u,s}^*$ where one will be stable $|DP(y_s^*)| < 1$ and other unstable $|DP(y_u^*)| > 1$ respectively. The left hand side curves of the Figure 20 are the derivative of the fixed points for the first subdomain, where their derivative have always positive values.

- **Second subdomain** $[y_{tan}^2, \bar{y}_{res}]$**:**
  In Figure 14 the second subdomain corresponds to the right hand side segments. For the second subdomain only one fixed point can exist for some values of Δ_y and can be stable or unstable depending on Δ_y. Figure 16 shows how the map P with fixed $V_{res} = -0.05$ is displaced for 3 different values of $\Delta_y = 0, 0.123, \Delta_y^{max}$. When $\Delta_y = 0.123$ an unstable y_u^* exists on the second subdomain and when $\Delta_y = \Delta_y^{max}$ a stable y_s^* exists.

  The right hand side curve of Figure 17 corresponds to the fixed points for the second subdomain for three $V_{res} = -0.04, -0.05, -0.06$. A zoom is in Figure 19 where the fixed point start being unstable and finish being stable. The right hand side curves in the Figure 20 correspond to the derivative of the fixed point of the second subdomain. Since its derivative have always negative values smaller than -1, the unstable fixed point is a **flip repeller**.
  It is important to express here that the second subdomain shows a transition from an unstable to stable fix point with a derivative going from $DP(y_u^*) < -1$ to $DP(y_s^*) > -1$. These kind of transitions led to period doubling [5],[6]. In the next section we talk about period doubling found in this system.

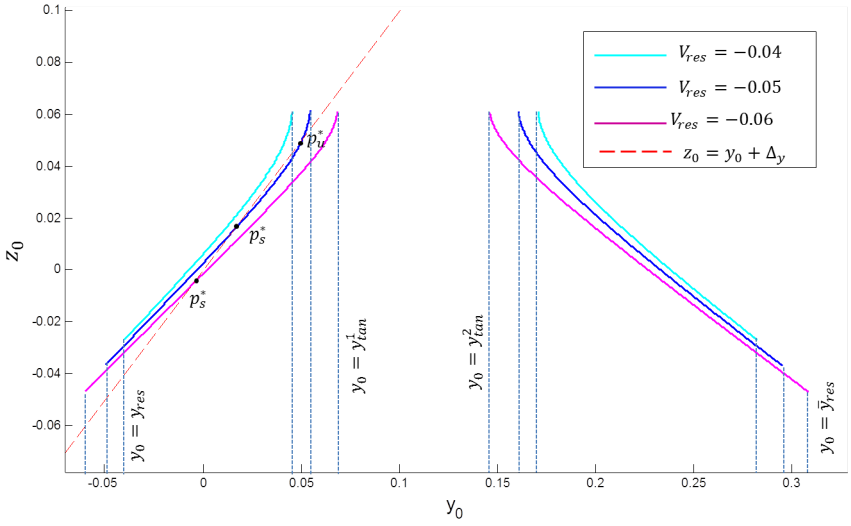

FIG. 14. **Case 3** mapping plot with $V_{res} = -0.04, -0.05, -0.06$, and fixed $\Delta_y = 0.013$

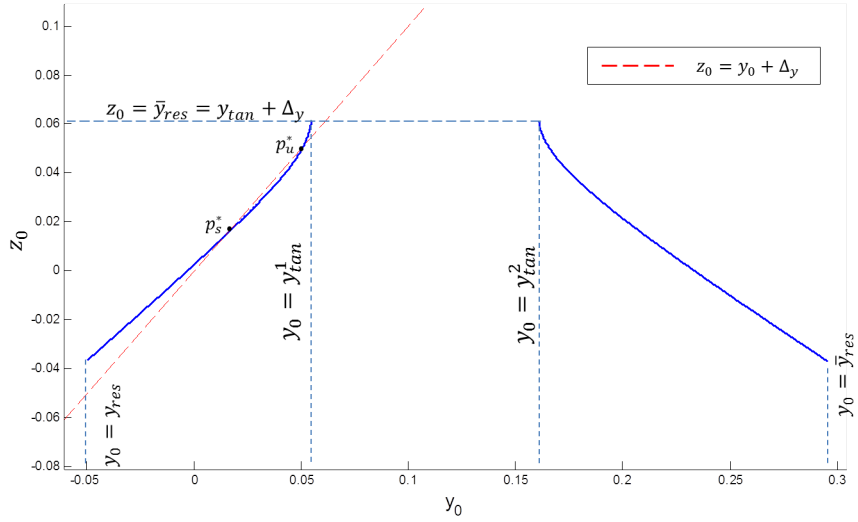

FIG. 15. **Case 3** mapping plot with $V_{res} = -0.05$, and fixed $\Delta_y = 0.013$

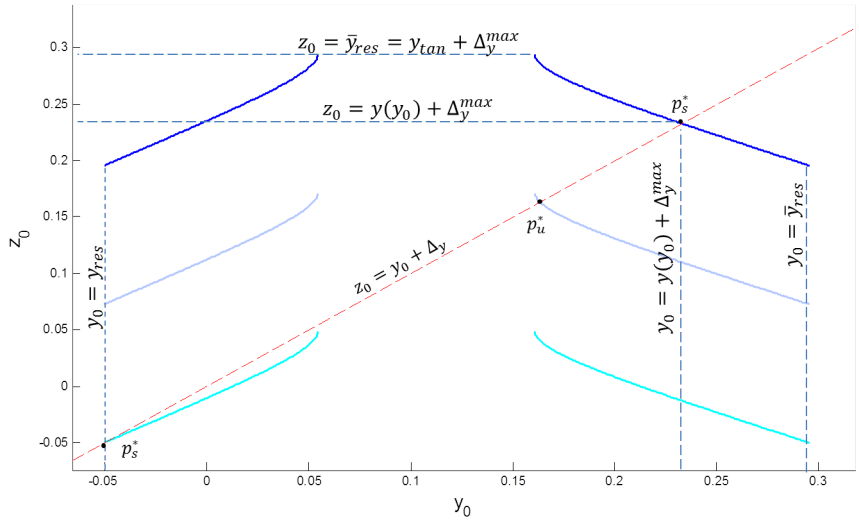

FIG. 16. **Case 3** mapping plots with $V_{res} = -0.05$, with different $\Delta_y = 0, 0.123, \Delta_y^{max}$; and where $\Delta_y^{max} = 0.246$

2.4.4. **Period Doubling.**

In the second subdomain we have a repelling fixed point y_u^* with derivative smaller than -1 (*flip repeller*) moving to a stable fixed pointy y_s^* with $DP(y_s^*) > -1$. This suggests that a period doubling bifurcation can happen. In this section we compute a map $\tilde{P}(y_0) = P(P(y_0))$, for a fixed $V_{res} = -0.05$ and for four different

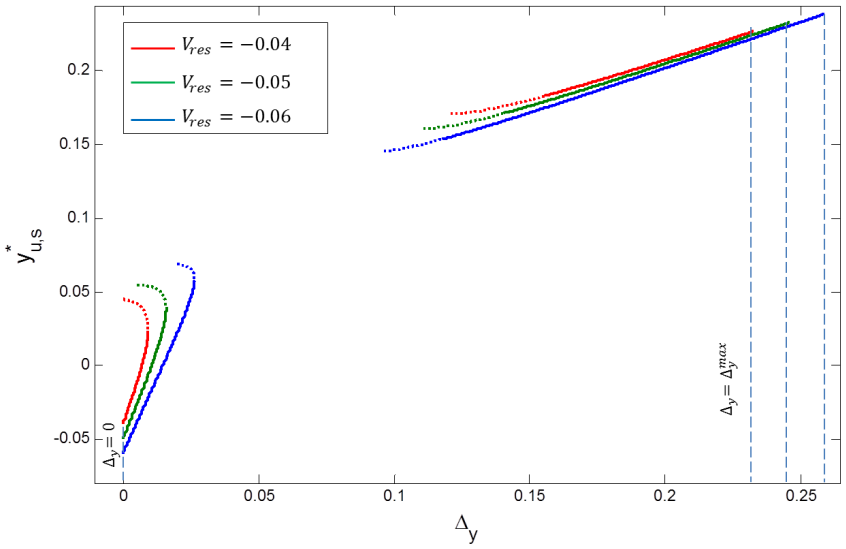

FIG. 17. **Case 3** fix point diagram with $V_{res} = -0.04, -0.05, -0.06$ respect to Δ_y, unstable y_u^* fix points are dotted lines, stable fix point y_s^* continuous line.

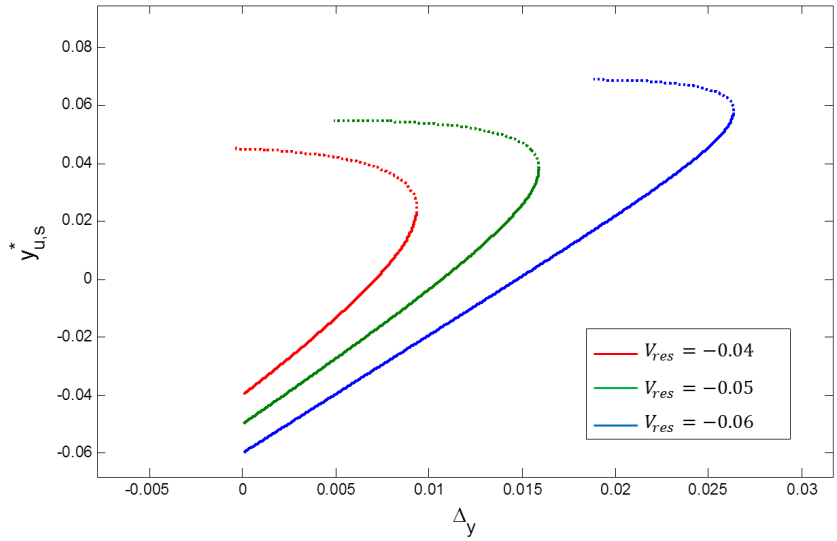

FIG. 18. **Case 3** First subdomain fix point diagram with $V_{res} = -0.05$ respect to Δ_y, unstable y_u^* fix points are dotted lines, stable fix point y_s^* continuous line.

$\Delta_y \leq \Delta_y^{max}$ values. Our goal is to understand the nature of the map $\tilde{P}$ and its dynamics with respect to Δ_y. We compute the period-2 fixed points by solving the

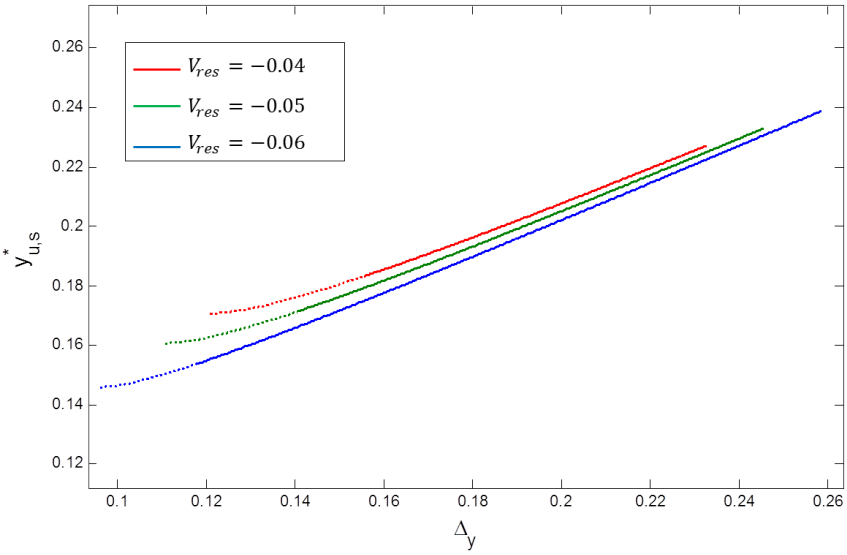

FIG. 19. **Case 3** Second subdomain fix point diagram with $V_{res} = -0.05$ respect to Δ_y, unstable y^*_u fix points are dotted lines, stable fix point y^*_s continuous line.

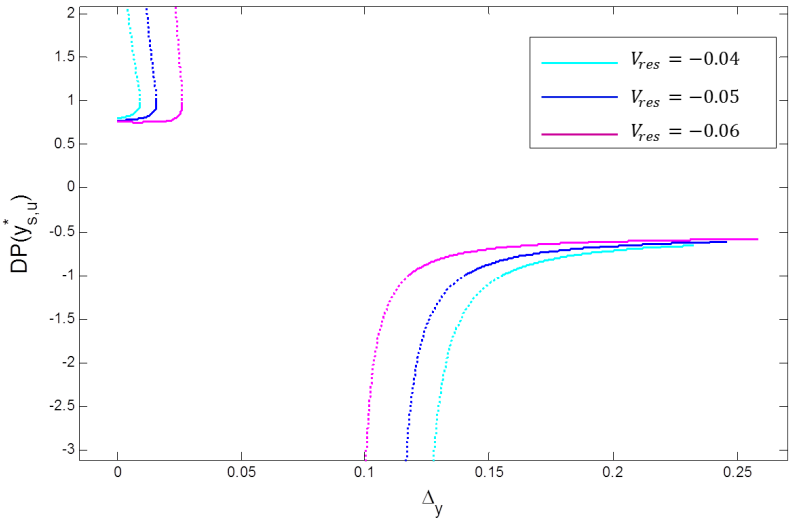

FIG. 20. **Case 3** derivative of fixed points with $V_{res} = -0.04, -0.05, -0.06$

equation $\tilde{P}(y_0) = P^2(y_0) = P(P(y_0)) = y_0$.

Hence using (89) we have:

(93)
$$\tilde{P}(y_0) = P^2(y_0) = P(P(y_0)) = m(m(y_0, t_{f_1}(y_0)) + \Delta_y, t_{f_2}(m(y_0, t_{f_1}(y_0)) + \Delta_y)) + \Delta_y$$

and recall from (87), that the time $t_{f_1}(y_0)$ verifies $N(y_0, t_{f_1}) = 0$, and $t_{f_2}(m(y_0, t_{f_1}(y_0)) + \Delta_y)$ verifies $N(t_{f_2}, m(y_0, t_{f_1}) + \Delta_y) = 0$.

In Figure 21 we show the iterates of the map $\tilde{P}$ and P with $V_{res} = -0.05$ and $\Delta_y = \Delta_y^{max} = 0.2456$. The map $\tilde{P}$ and P has only one intersection with the bisecting line $z_0 = y_0$ for this $\Delta_y = \Delta_y^{max}$, which remains to be the same stable fixed point y_s^* of P.

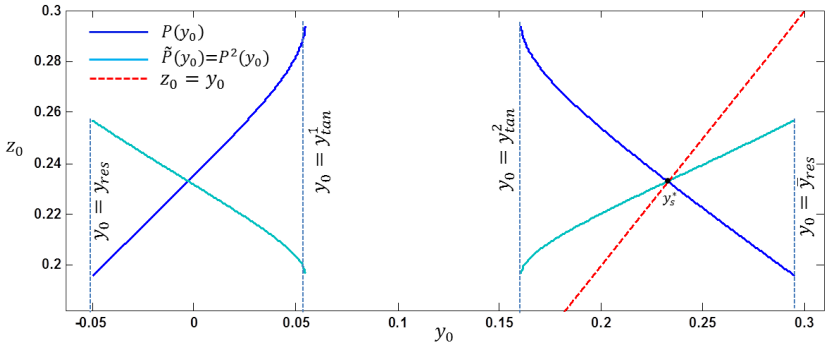

FIG. 21. **Case 3** map P and P^2 for $V_{res} = -0.05$ with $\Delta_y = \Delta_y^{max} = 0.2456$

In the Fig.22 we show the maps $\tilde{P}$ and P with $V_{res} = -0.05$ for other $\Delta_y \leq \Delta_y^{max}$ values showing only the second subdomain. Figure 22 (a) shows the map $\tilde{P}$ with Δ_y^{max}, and Fig. 22 (b) with $\Delta_y = 0.1537$ it shows three intersections with $z_0 = y_0$ generating two $y_u^{*(2)}$ unstable fix points of period-2 and the remaining y_s^* period-1 stable fix point. Notice that the map $\tilde{P}$ of side Fig.22(b) is not defined in the whole second subdomain $[y_{tan}^2, \bar{y}_{res}]$ as the map P. This happens because the full domain D_P for case 3 is disconnected.

In Figure 23 (a) the maps for P and $\tilde{P}$ for $\Delta_y = 0.145$, and in 23 (b) a zoom of it. One can see that both unstable fixed points $y_u^{*(2)}$ start to approach as Δ_y varies the stable fix point y_s^* of period-1.

In the Figure 24 (a) the maps P and $\tilde{P}$ for $\Delta_y = 0.13$ only the y_u^* unstable fix point of period-1 appears, while the unstable $y_u^{*(2)}$ fixed points of period-2 have already disappeared. Also the domain of $\tilde{P}$ has decreased its extension over the second subdomain $[y_{tan}^2, \bar{y}_{res}]$. This can be explained because not all the values of $y_0 \in [y_{tan}^2, \bar{y}_{res}]$ have an image defined in $\tilde{P}(y_0)$ as Δ_y is increased. The values of $y_0 \in [y_{tan}^2, \bar{y}_{res}]$ that do not have an image through $\tilde{P}$ indicate that after the first

iteration $P(y_0) \in [y^1_{tan}, y^2_{tan}]$, that is they fall in the domain of attraction of the stable focus.

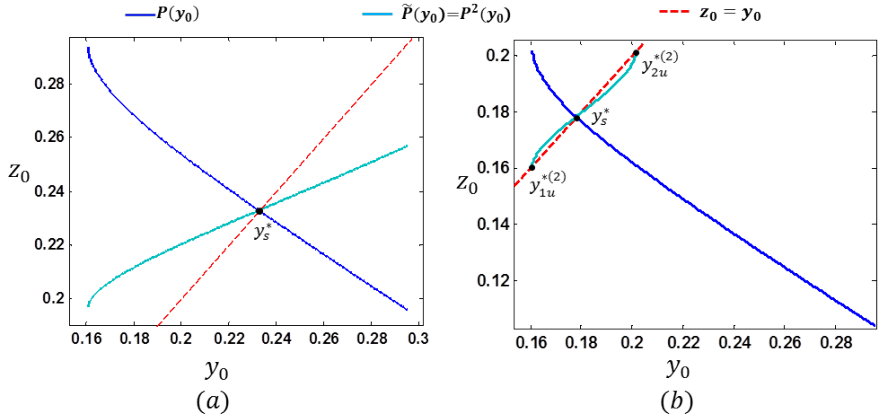

FIG. 22. **Case 3** side (a) the map P and P^2 for $V_{res} = -0.05$ with $\Delta_y = \Delta_y^{max}$, for side (b) we have $V_{res} = -0.05$ with $\Delta_y = 0.1537$

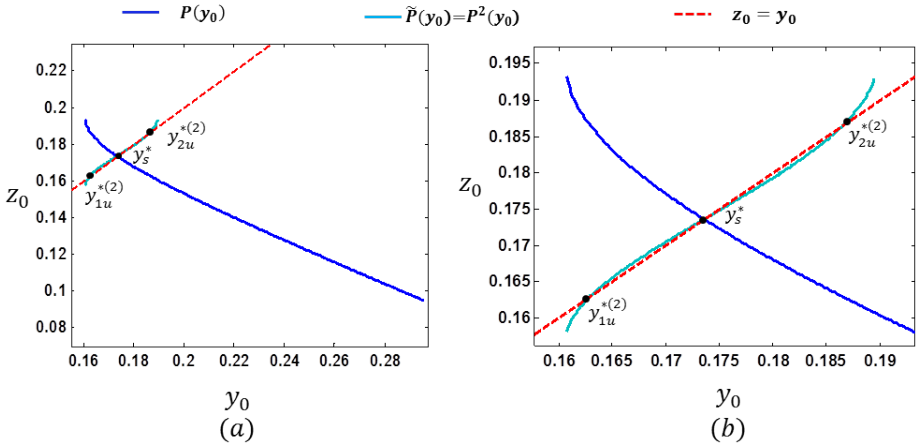

FIG. 23. **Case 3** side (a) the map P and P^2 for $V_{res} = -0.05$ with $\Delta_y = 0.145$, for side (b) we show a zoom

The bifurcation diagram for the fixed points of the map $\tilde{P}$ was computed by solving $\tilde{P}(y_0) = y_0$ for $V_{res} = -0.05$. A subcritical pitchfork bifurcation happens as $\Delta_y \leq \Delta_y^{max}$ is increased on the second subdomain, see Fig. 25. There is an unique branch of period-1 unstable fixed point y_u^* which bifurcates into three different branches, two branches are *unstable* fixed points of period-2 $y_u^{*(2)}$ that exist around *stable* branch of fixed points y_s^* of period-1. Notice that the branches of period-2 $y_u^{*(2)}$ finish before the stable branch y_s^*. Other bifurcations for $V_{res} = -0.05$, $V_{res} = -0.04$ and $V_{res} = -0.03$ are shown in Fig. 26.

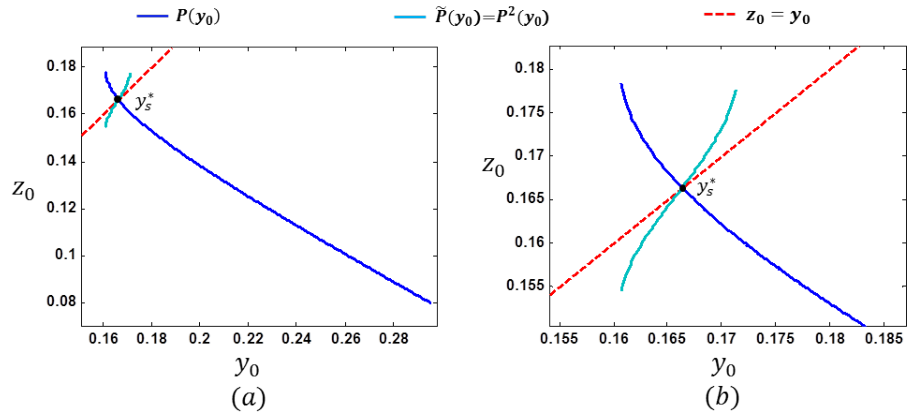

FIG. 24. **Case 3** side (a) the map P and P^2 for $V_{res} = -0.05$ with $\Delta_y = 0.13$, for side (b) we show a zoom

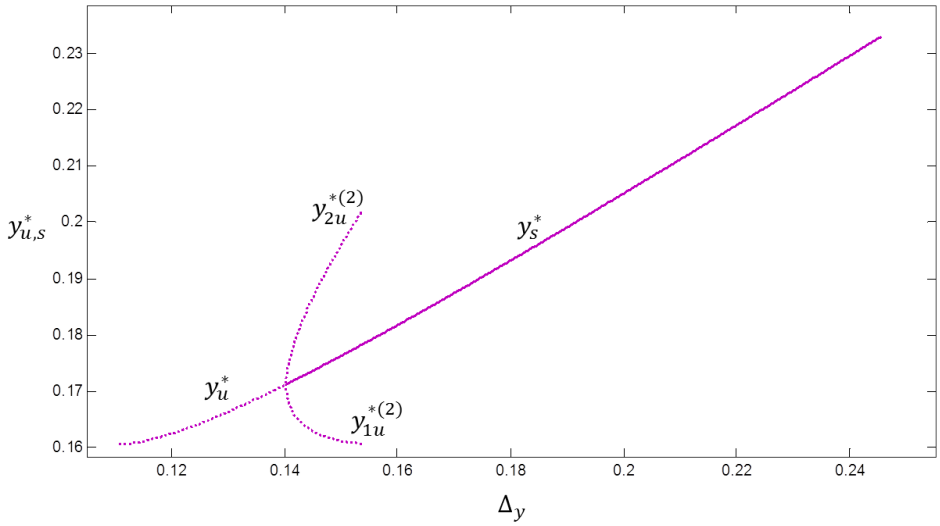

FIG. 25. **Pitchfork** Bifurcation for $V_{res} = -0.05$ respect to Δ_y, unstable branch are dotted lines, stable branch is a continous line.

2.4.5. **Fixing Δ_y and varying $V_{res} \leq V_{res}^*$.** In this section we proceed to fix Δ_y value and vary V_{res} along the interval $V_{res} \in [-0.13, V_{res}^*]$. We show the respective curves found by varying V_{res} as the main parameter. We fixed different values for Δ_y and vary V_{res} as shown on Fig. 27. However the red line does not have a fixed Δ_y. This line defines the fixed points for all the Δ_y^{max}, where Δ_y^{max} varies with respect to the V_{res} value and defines an upper limit for the other curves with fixed Δ_y. The black line is defined by fixing $\Delta_y = 0$ and varying V_{res}. It is clear that curves are continuous and there is a unique fixed point for each value of the parameter $V_{res} < V_{res}^*$ when fixing $\Delta_y < \Delta_y^{max}$.

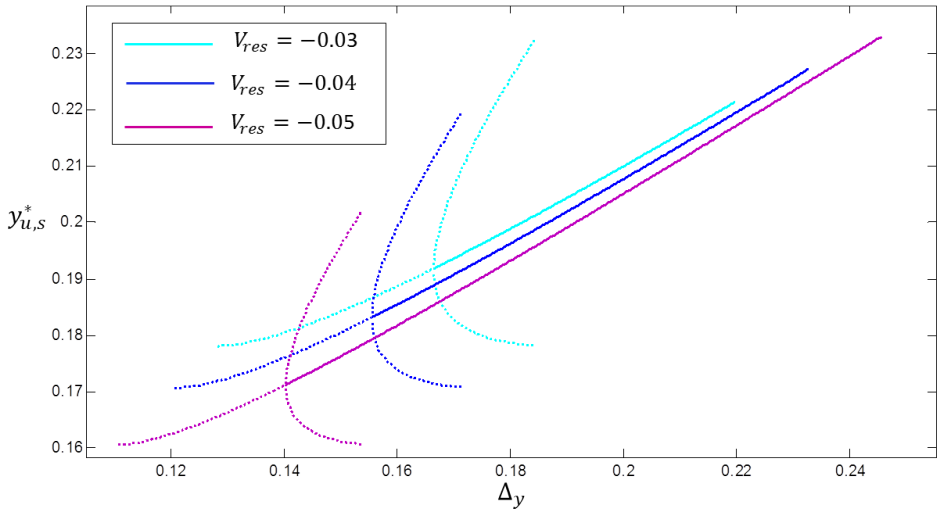

FIG. 26. **Pitchfork** Bifurcation for $V_{res} = -0.05, -0.04, -0.03$ respect to Δ_y, unstable branch are dotted lines, stable branch is a continous line.

In Figure 28 we show the derivative of the fixed point. It always have $|DP(y_s^*)| < 1$ confirming that fixed points with $V_{res} = [0.13, V_{res}^*]$ are stable for any $\Delta_y = [0, \Delta_y^{max}]$.

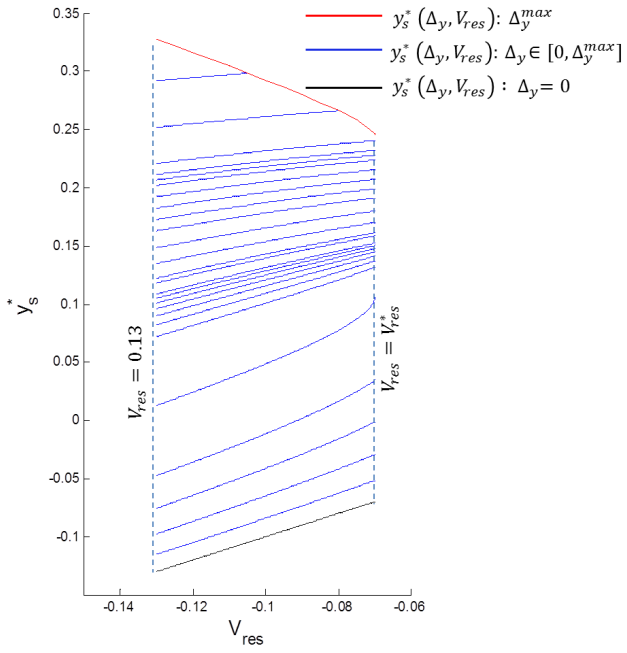

FIG. 27. **Case 1** fixed points diagram with parameter V_{res} for ten different $\Delta_y < \Delta_y^{max}$ values. The dotted line belongs to $\Delta_y = 0$

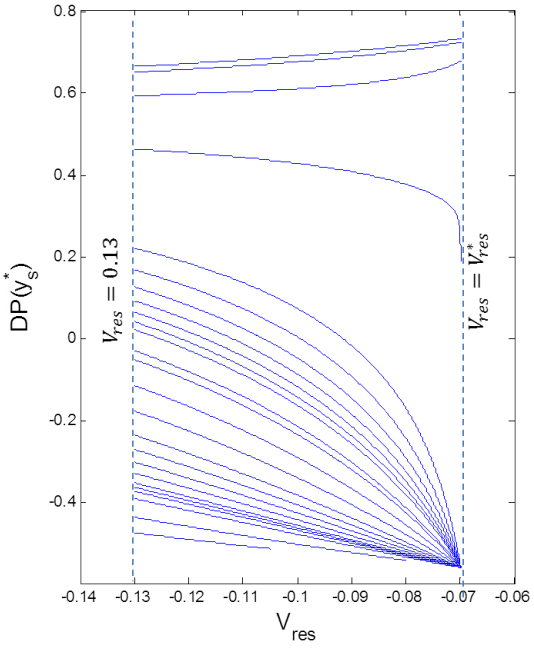

FIG. 28. **Case 1** derivative of fixed points.

2.4.6. **Fixing Δ_y and varying $V_{res} > V^*_{res}$.**

We proceed as before. We fixed a Δ_y value and vary V_{res} along the interval $V_{res} \in [V^*_{res}, x^*_e]$ until we reach the focus. Some curves undergo a bifurcation while others showed to have just one fixed point stable or unstable.
Each curve in Fig. 29 corresponds to a different fixed Δ_y and V_{res} varying in $[V^*_{res}, x^*_e]$. Again here the red line does not have a fixed Δ_y. It is defined by all the Δ^{max}_y, where Δ^{max}_y varies with respect to V_{res}. Also it defines an upper limit for the other curves with fixed Δ_y. The black line is the bisecting line $x = y$. The focus was also added to this figure in order to make clearer. The curves of the fixed points are similar to the bifurcation diagram in Fig.17 but from another perspective. Lower lines defined below $y^*_{u,s} < y^*_{res}$ (which correspond to first subdomain) have a bifurcation, while the upper lines only have an unstable or stable fixed point (second subdomain).
Finally in Figure 30 the upper curves correspond to the derivatives of the fixed points on the second subdomain $y^*_{u,s} > y^*_{res}$, and lower curves to the branches of the first subdomain $y^*_{u,s} < y^*_{res}$.

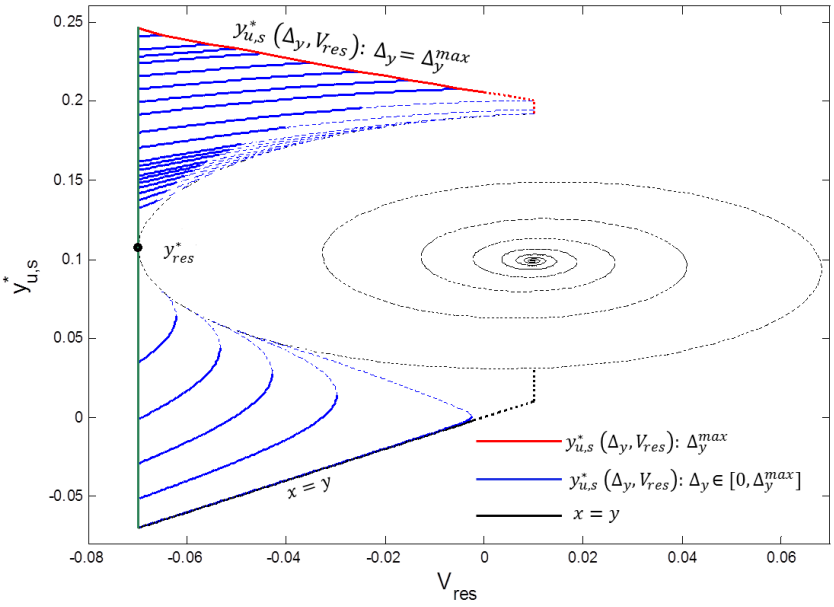

FIG. 29. **Case 3** fixed points diagram with parameter V_{res} for different $\Delta_y \in [0, \Delta_y^{max}]$ values, unstable fix points are dotted lines, stables are continous line.

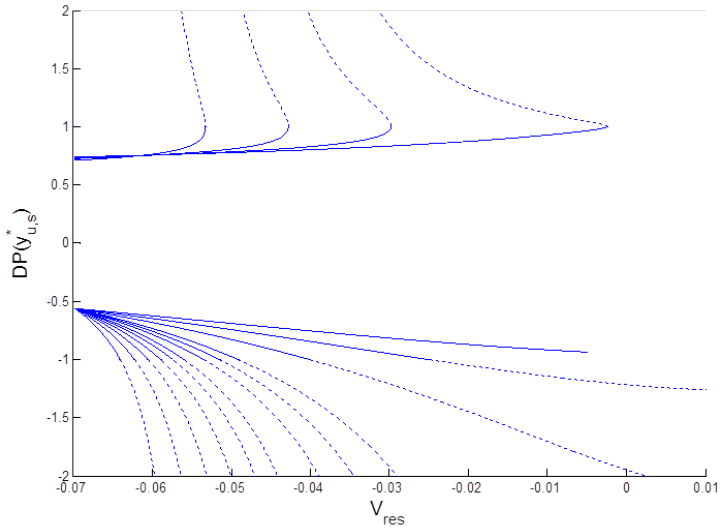

FIG. 30. **Case 3** Derivative of the fixed points.

2.4.7. **Conclusion for Fixed Points.** Summarizing then we have:

- Case 1 & Case 2: $V_{res} \in [-0.13, V^*_{res}]$
  (1) Only a stable fixed point y^*_s exists.

(2) This fixed point is defined on $y_s^* \in [y_{res}, y_{res} + \Delta_y^{max}] \subset D_P$ for a respective $\Delta_y \in [0, \Delta_y^{max}]$ and $V_{res} \in [-0.13, V_{res}^*]$.
(3) We also define $\Sigma_{st} \subset \Sigma_0$ as the domain of attraction of the fixed point y_s^*. For this case we have always $\Sigma_{st} = D_P = [y_{res}, \bar{y}_{res}]$.

- Case 3 $V_{res} \in (V_{res}^*, x_e^*]$
  (1) When $\Delta_y > 0$ and small a stable fix points y_s^* appears in the first subdomain $[V_{res}, y_{tan}^1]$ see (82). In this case the domain of attraction of the fixed points Σ_{st} is $\Sigma_{st} = [V_{res}, y_{tan}^1] \cup [y_{tan}^2, \bar{y}_{res}] = D_P$.
  (2) An unstable fix point y_u^* appears in the first subdomain for some value of Δ_y and, as Δ_y increases, y_s^* and y_u^* meet in a saddle-node bifurcation and disappear. The domain of attraction of the stable fix point y_s^* is $\Sigma_{st} = [y_{res}, y_u^*) \cup (y_u^p, \bar{y}_{res}]$. Here the point $p_u^p = (V_{res}, y_u^p)$ denotes the intersection of the backward orbit through the point $p_u^* = (V_{res}, y_u^*)$ with the surface Σ_0. In this case $\Sigma_{st} \subset D_P$.
  (3) For some intermediate values of Δ_y the map P has no fixed points.
  (4) For larger values of Δ_y an unstable fix point y_u^* appears in the second subdomain $[y_{tan}^2, \bar{y}_{res}]$ see (82).
  (5) As Δ_y increases a period doubling bifurcation occurs. The fixed point becomes stable and an unstable periodic orbit $\{y_{1u}^{*(2)}, y_{2u}^{*(2)}\}$ of period-2 appears in the second subdomain. In this case the stability region $\Sigma_{st} = (y_{1u}^{*(2)}, y_{2u}^{*(2)}) \cup (y_{1u}^p, y_{2u}^p)$. Here the points $p_{1u}^p = (V_{res}, y_{1u}^p)$ and $p_{2u}^p = (V_{res}, y_{2u}^p)$ denotes the intersection of the orbit through the points $p_{1u}^* = (V_{res}, y_{1u}^*)$ and $p_{2u}^* = (V_{res}, y_{2u}^*)$ with the surface Σ_0 respectively.

2.4.8. **Two parameter (V_{res} and Δ_y) bifurcation analysis.**

In this section we assembled the fixed points from the three different cases by considering both parameters V_{res} and Δ_y. In Figure 31 is shown the chart of Δ_y vs. $y_{u,s}^*$ in the three cases. This figure gives an overview of the geometrical pattern defined by the lines where the saddle-node bifurcation happens only for case 3.
In Figure 32 we show $y_{u,s}^*$ vs. $V_{res} \in [0.13, x_e^*]$. V_{res}. In Fig. 33 we show a a 3D plot joining together $y_{u,s}^*$, V_{res} and Δ_y, taking the parameters as coordinates with $X = V_{res}$, $Y = \Delta_y$ and $Z = y_{u,s}^*$. The gap in the plot is for the stable focus position, and all the blue lines correspond to the stable fixed point y_s^* and the red lines to the unstable ones y_u^*. It is important to know that the periodic orbit through y_u^* separates the domain of attraction of the focus and the domain of attraction of the stable periodic orbit through y_s^* if the latter exists.

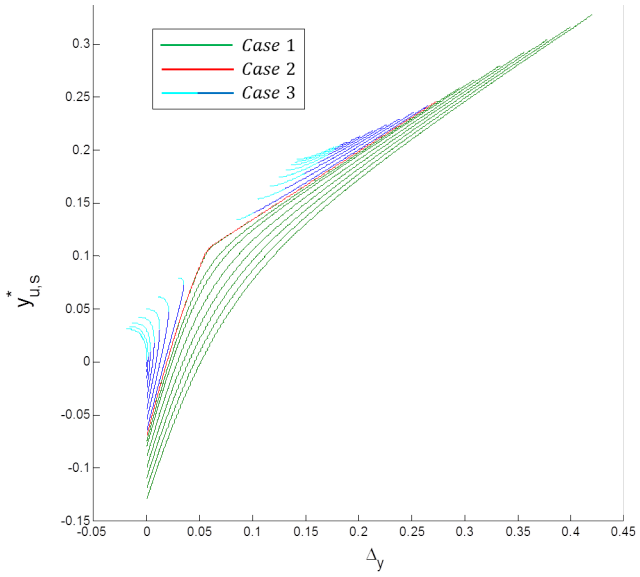

FIG. 31. **Case1, 2** and 3 shows fixed point respect to Δ_y, the light blue color denotes the unstable fixed points

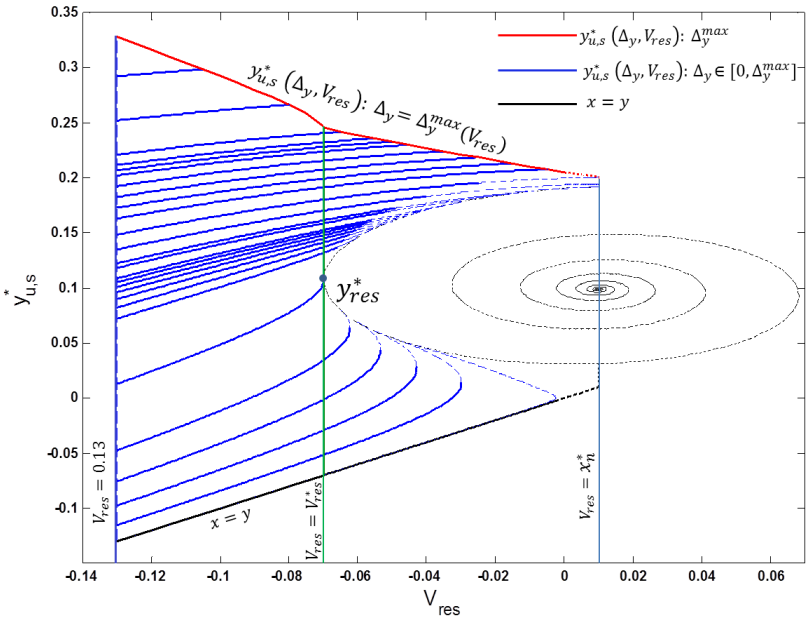

FIG. 32. **Case1, 2** and 3 shows fixed point respect to V_{res}

2.5. Conclusion for Periodic Orbits.

In this section we explain the nature of the periodic orbits and how they are linked to the fixed points previously found in the three cases. We take into consideration

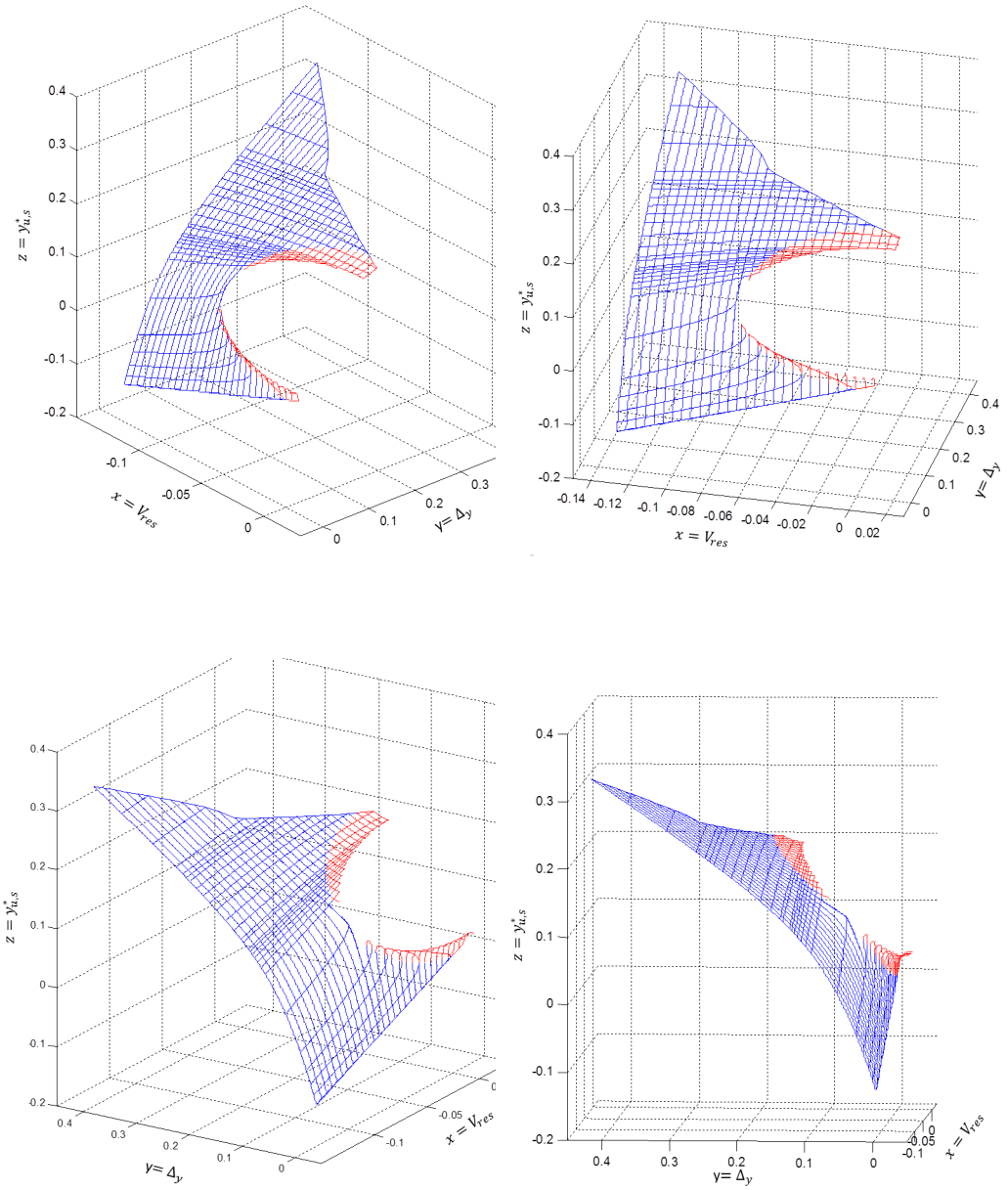

FIG. 33. A 3D dimensional chart for **Case1**, **2** and **3** showing $X = V_{res}$ $Y = \Delta_y$ $Z = y^*_{u,s}$, blue lines belong to stable fix points and red line to unstable ones.

case 1 and case 2 for a single explanation as there is no difference between their periodic orbits.

2.5.1. **Case 1 and Case 2.**

In this case a stable periodic orbit exists. This periodic orbit is defined by $\varphi_p(t) = \varphi(t, p_s^*)$ where $p_s^* = (V_{res}, y_s^*)$. Then $\varphi(t_1, p_s^*) \in \Sigma_1$ for $t_1 > 0$ and we apply the post-spike conditions which lead us to p_s^* again. As in this case $\Sigma_{st} = D_P$ the basin of attraction of this periodic orbit is given by the region between the orbits of the point p_{res}, the orbit of the point p_{tan} and the line $x = y$ see fig. 34. The rest of points are attracted by the focus z_e^*. So in this case the system shows bistability between a fixed point and periodic orbit.

An example is shown in Figure 35, where $\varphi(t, z_0)$ with $z_0 \in D_P$ is taken, then we reach Σ_1 and the post-spike conditions are applied. This solution converges to φ_p as $t \to \infty$. The variables x=*membrane potential* and y=*dynamic threshold* of the solution of Fig. 35 are shown in Fig. 36 as functions of time. The spikes are generated everytime the membrane potential reaches the dynamic threshold $x = y$, and x is reset to V_{res} after every spike.

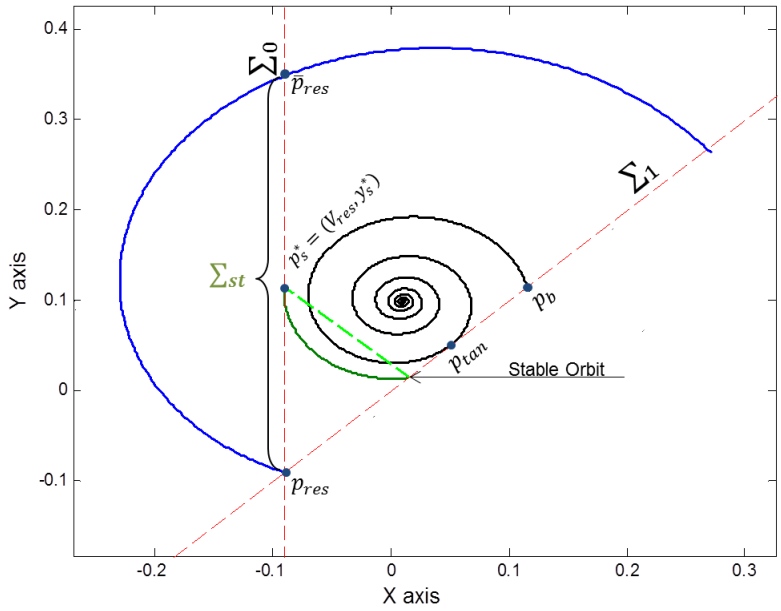

FIG. 34. **Case 1** An stable orbit with values $V_{res} = -0.09$, $\Delta_y = 0.1$ and $y_s^* = 0.1146$

2.5.2. **Case 3 and periodic orbits.**

When $\Delta_y > 0$ and small a stable periodic orbit $\varphi(t, p_s^*)$, where $p_s^* = (V_{res}, y_s^*)$, appears. In Figure 37 the stable periodic orbit $\varphi(t, p_s^*)$ defined with $V_{res} = -0.05$ and $\Delta_y = 0.004$ is shown. Again in this case we have that $\Sigma_{st} = D_P$, therefore the basin of attraction of this periodic orbit is given by the region between the orbits of the point p_{res}, the orbit of the point p_{tan} and the line $x = y$ see fig. 37. The rest of points are attracted by the focus z_e^*. Any solution $\varphi(t, z_0)$ with $z_0 \in D_P$ leads to the stable periodic orbit φ_p. Thus, there are not finite sustained spiking for this case.

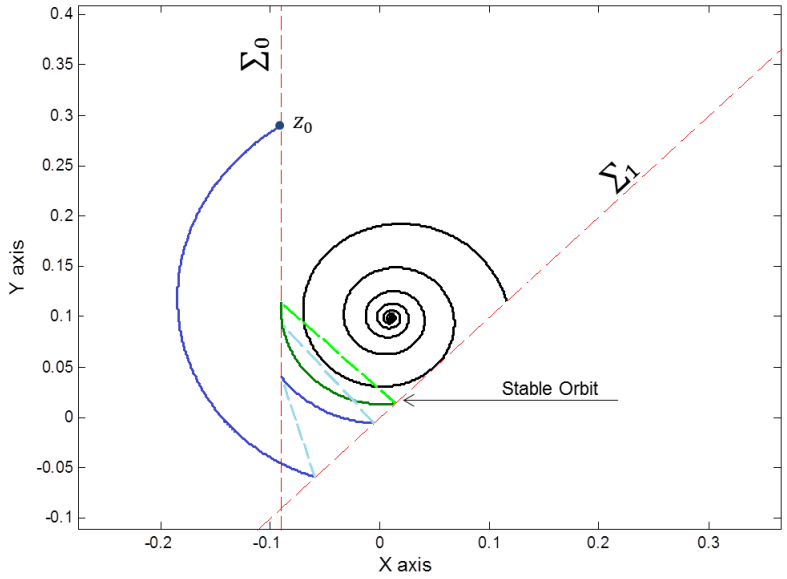

FIG. 35. **Case 1** A solution φ intersect Σ_1 in different places as it converges to the stable orbit. Values: $V_{res} = -0.09$, $\Delta_y = 0.1$ and $y_s^* = 0.1146$

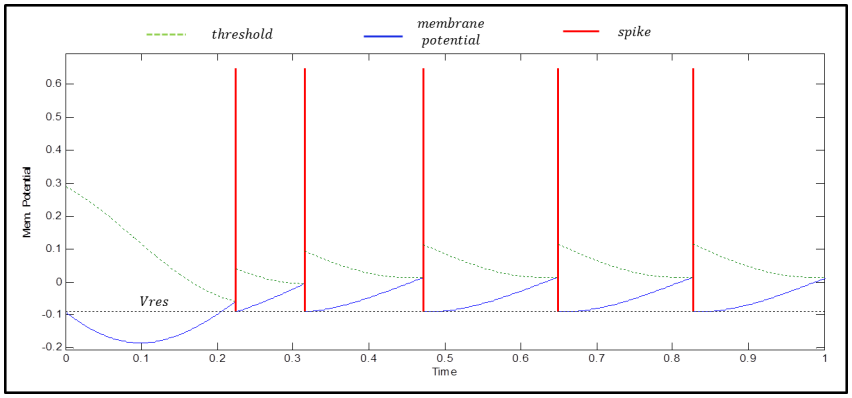

FIG. 36. **Case 1** time-Voltage plot. The convergence to a stable orbit translates like an infinite spiking as $t \to \infty$. Values: $V_{res} = -0.09$, $\Delta_y = 0.1$ and $y_s^* = 0.1146$

For some Δ_y an unstable periodic orbit appears $\varphi(t, p_u^*)$, where $p_u^* = (V_{res}, y_u^*)$, in the first subdomain. Then as Δ_y increases the stable and unstable periodic orbits $\varphi(t, p_s^*)$ and $\varphi(t, p_u^*)$ meet in a saddle-node bifurcation and disappear. In Figure 38 we show the stable and unstable periodic orbits, the parameters used are: $V_{res} = -0.05$ and $\Delta_y = -0.015$. As in this case $\Sigma_{st} = [y_{res}, y_u^*) \cup (y_u^p, \bar{y}_{res}]$, the basin of attraction of the stable periodic orbit is given by the region between the orbits of the point p_{res}, the unstable periodic orbit and the line $x = y$ see Figure 38. The rest of points are attracted by the focus z^*e.

The periodic orbits from Fig. 38 are shown in Fig. 39 with a solution $\varphi(t, z_0)$ with $z_0 \in \Sigma_{st}$ approaching the stable periodic orbit φ_p. Figure 40 reflects the variables in a V plot.

Figure 41 shows a solution $\varphi(t, z_0)$ with $z_0 \notin \Sigma_{st}$ being repelled from the unstable periodic orbit and attracted to the focus $\varphi(t, z_0) = z_e^*$ as $t \to \infty$. The Fig. 42 reflects the variables in a $V - t$ plot. Notice that here finite spiking is possible.

For some intermediary values of Δ_y we do not have any periodic orbit. All solutions $\varphi(t, z_0)$ converges to the stable focus as $t \to \infty$.

For larger values of Δ_y only an unstable periodic orbit $\varphi(t, p_u^*)$ exists in the second subdomain (see Fig. 43). Therefore all solutions $\varphi(t, z_0)$ with $z_0 \neq p_u^*$ converges to the stable focus as $t \to \infty$. In Figure 44 we show a solution $\varphi(t, z_0)$ repelled from the unstable periodic orbit and finally approaching the focus.

As Δ_y increases a period doubling bifurcation occurs. The unstable periodic orbit becomes stable and an unstable periodic orbit with two spikes per period appears. Now the basin of attraction of the stable periodic orbit is the region delimited by the unstable periodic orbit which contains the stable one, see Fig. 45. The rest of the points converges to the focus.

An example of a solution $\varphi(t, z_0)$ with $z_0 \notin \Sigma_{st}$ is in Fig. 46. The solution is repelled from the unstable periodic orbit and creates four spikes before converging to the focus. The reflection in a $V - t$ plot is in Fig. 47.

Finally for the higher values of $\Delta_y \leq \Delta_y^{max}$ the unstable periodic orbit dissapears and the stable periodic orbit remains. Again in this case we have that $\Sigma_{st} = D_P$, therefore the basin of attraction of this periodic orbit is given by the region between the orbits of the point p_{res}, the orbit of the point p_{tan} and the line $x = y$. The rest of points are attracted by the focus z_e^*. Any solution $\varphi(t, z_0)$ with $z_0 \in D_P$ leads to the stable periodic orbit φ_p. Thus, there are not finite sustained spiking for this case, see Fig.48.

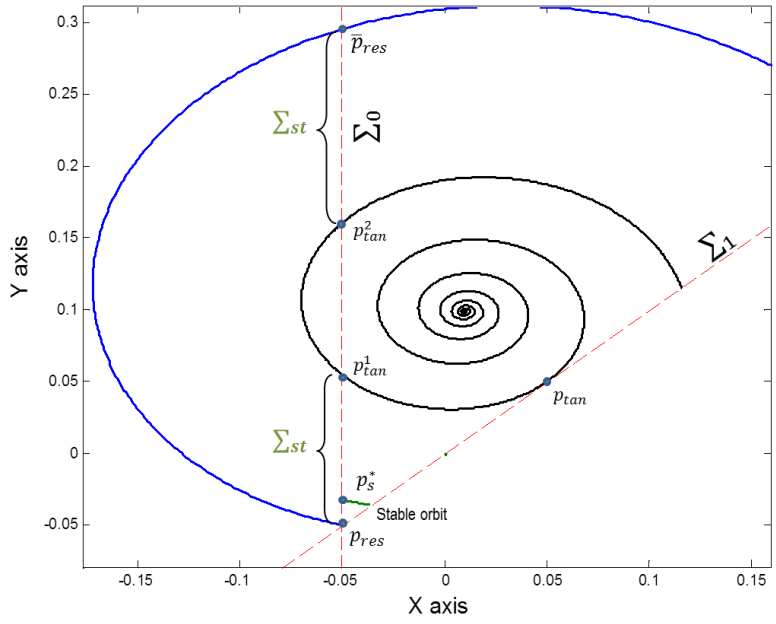

FIG. 37. **Case 3** The stable orbit defined on the first subdomain with values: $V_{res} = -0.05$, $\Delta_y = 0.004$ with $y^*_s = -0.032$

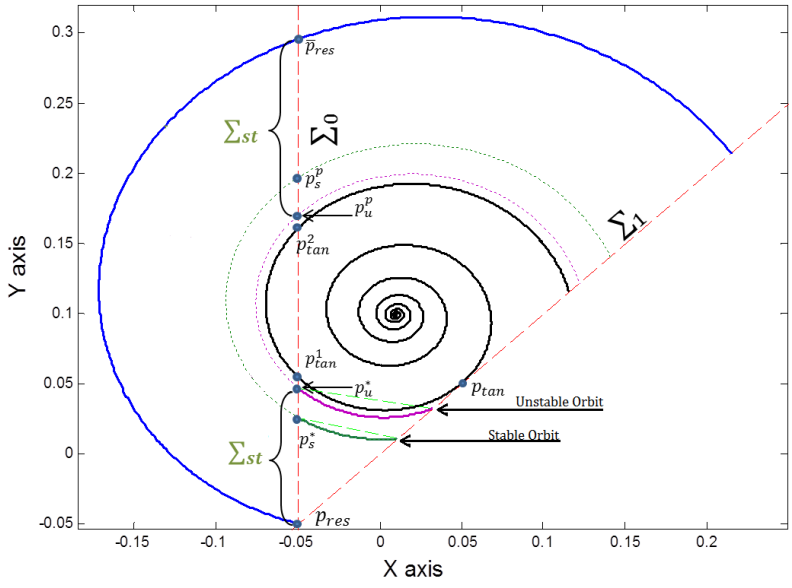

FIG. 38. **Case 3** The stable orbit and unstable orbit defined on the first subdomain. Values: $V_{res} = -0.05$, $\Delta_y = 0.015$, $y^*_u = 0.0468$ and $y^*_s = 0.0255$

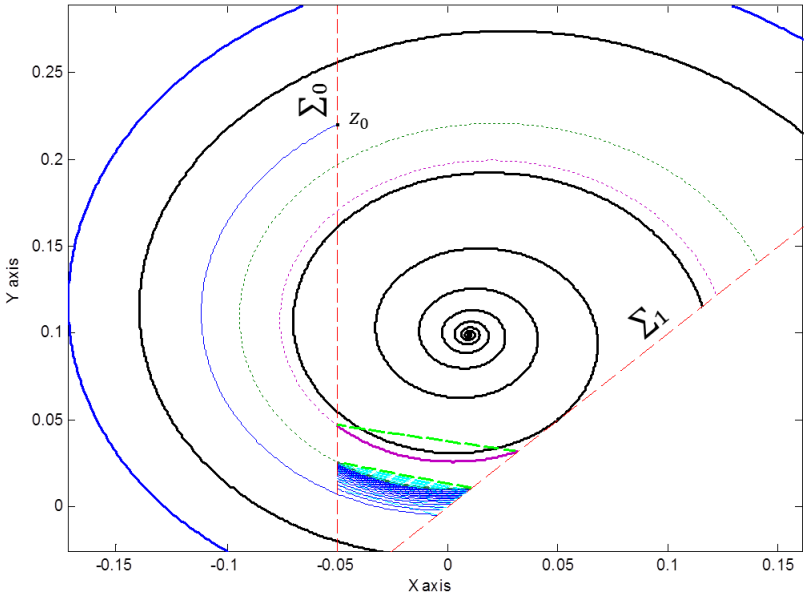

FIG. 39. **Case 3** A solution φ intersects Σ_1 in different places as it reaches the stable orbit from the first subdomain. Values: $V_{res} = -0.05$, $\Delta_y = 0.015$ and $y_s^* = 0.0255$

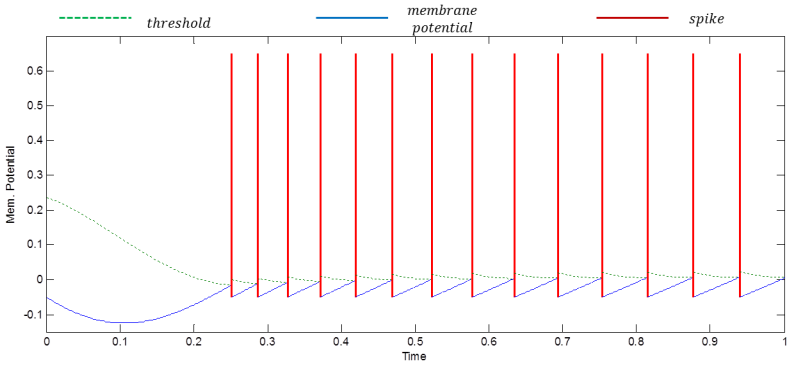

FIG. 40. **Case 3** time-Voltage plot. The convergence to the stable orbit translates as an infinite spiking as $t \to \infty$. Values: $V_{res} = -0.09$, $\Delta_y = 0.015$ and $y_s^* = 0.0255$

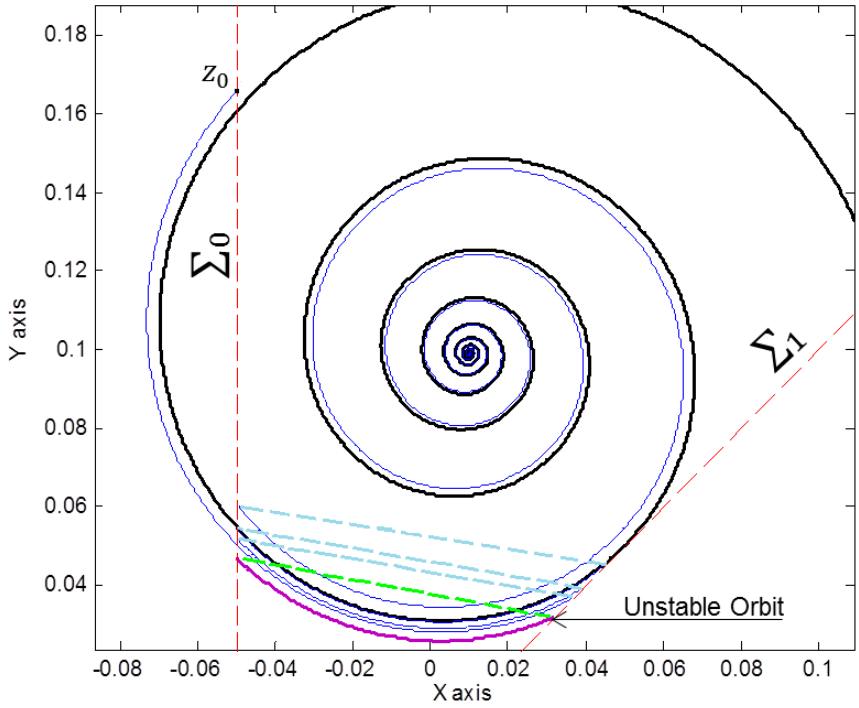

FIG. 41. **Case 3** A solution $\varphi(t, z_0)$ with $z_0 \notin \Sigma_{st}$ is repelled from the unstable orbit. φ intersect Σ_1 three times and then arrives to the focus. This process produce three spikes. Values: $V_{res} = -0.05$, $\Delta_y = 0.015$ and $y_u^* = 0.0468$

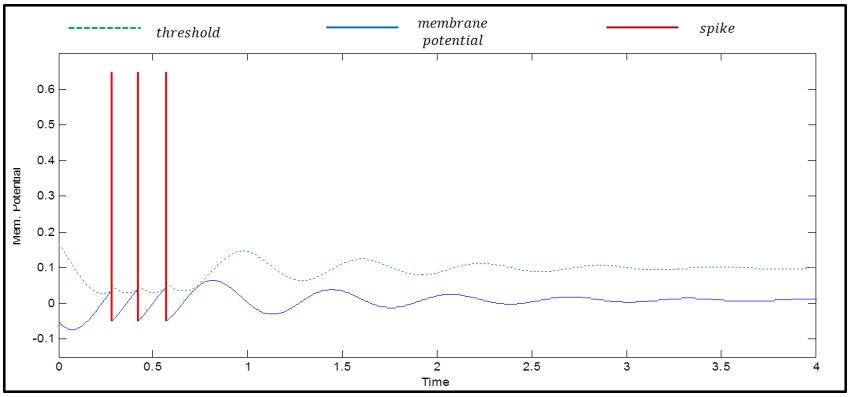

FIG. 42. **Case 3** time-Voltage plot. Three spikes are generated before arriving to the resting state. Values: $V_{res} = -0.05$, $\Delta_y = 0.015$ and $y_u^* = 0.0468$

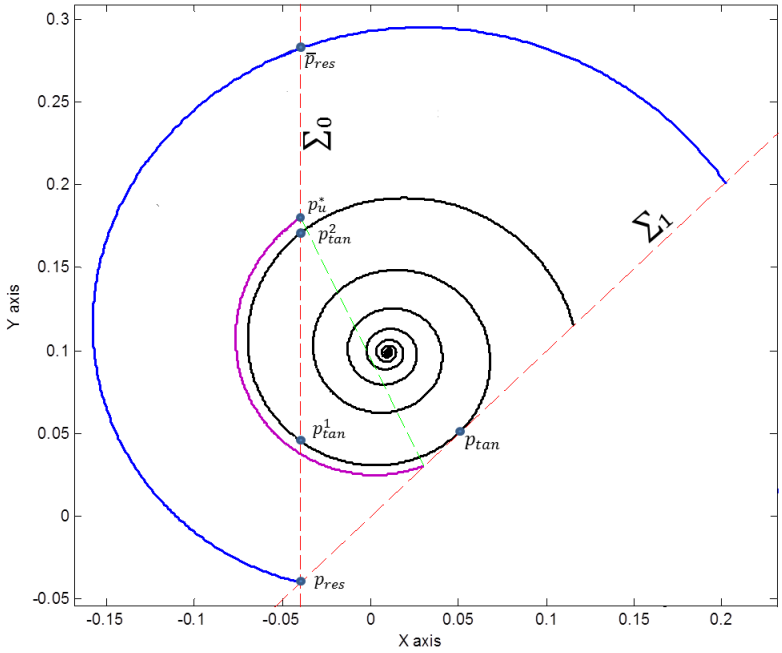

FIG. 43. **Case 3** The unstable orbit defined on the second subdomain for the values: $V_{res} = -0.04$,$\Delta_y = 0.15$ and $y_u^* = 0.1804$

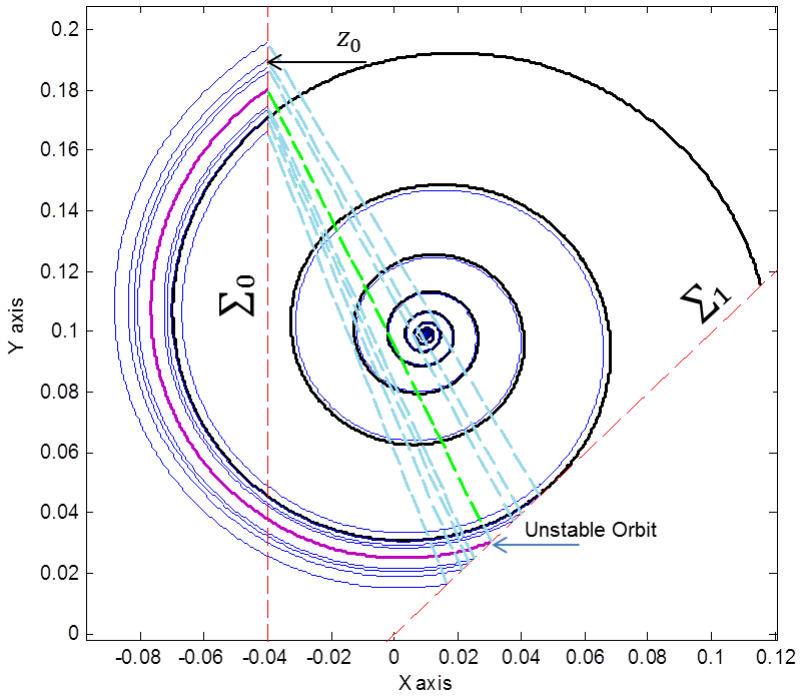

FIG. 44. **Case 3** A solution $\varphi(t, z_0)$ with $z_0 \notin \Sigma_{st}$ is repelled intersecting Σ_1 several times and then arrives to the focus. Values: $V_{res} = -0.04$,$\Delta_y = 0.15$ and $y_u^* = 0.1804$

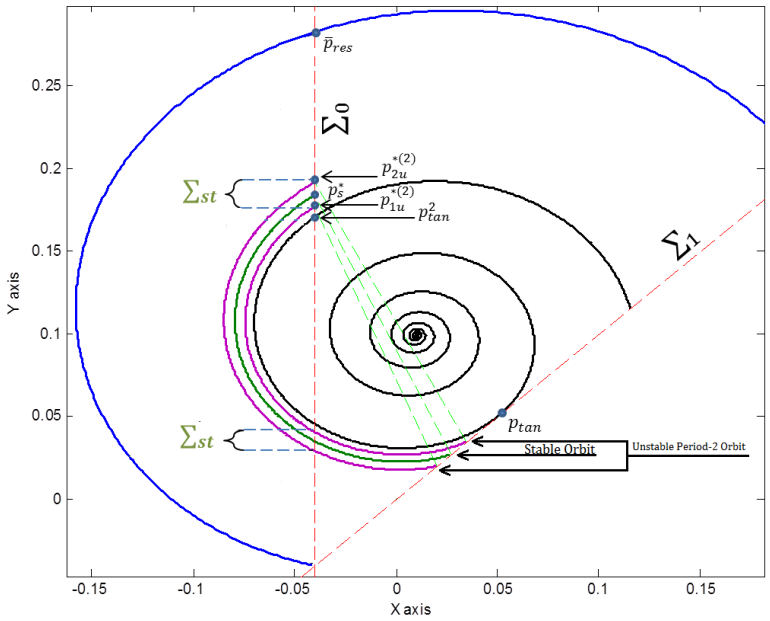

FIG. 45. **Case 3** Three periodic orbits are defined. One stable orbit of period-1 and two unstable periodic orbits of period-2. Values: $V_{res} = -0.05$, $\Delta_y = 0.157$, $y_{1u}^{*(2)} = 0.1772$, $y_s^* = 0.184$ and $y_{2u}^{*(2)} = 0.1915$

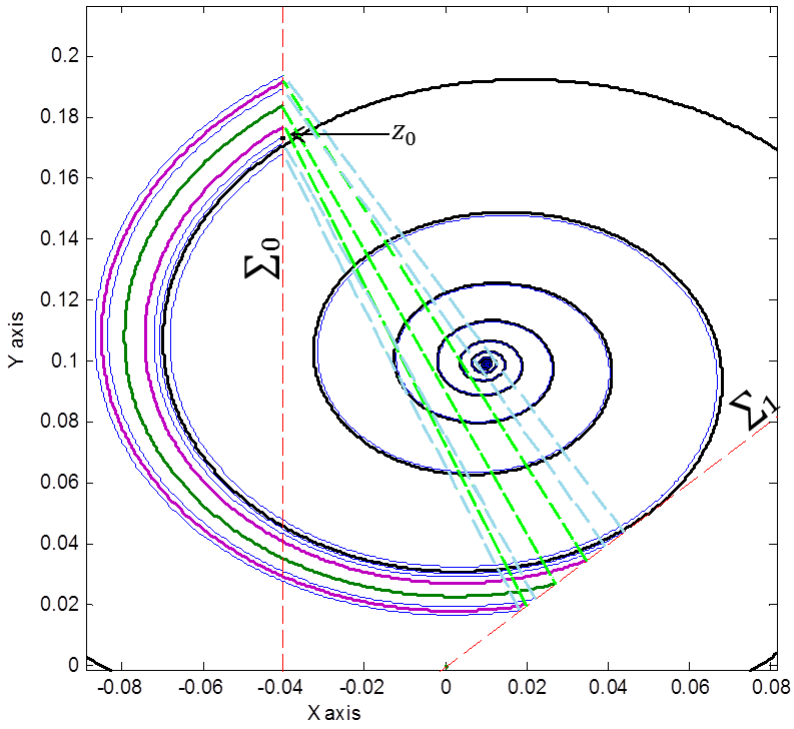

FIG. 46. **Case 3** A solution $\varphi(t, z_0)$ with $z_0 \notin \Sigma_{st}$ is repelled. In this picture φ reaches Σ_1 four times and then arrives to the focus. Same parameters from Fig. 48

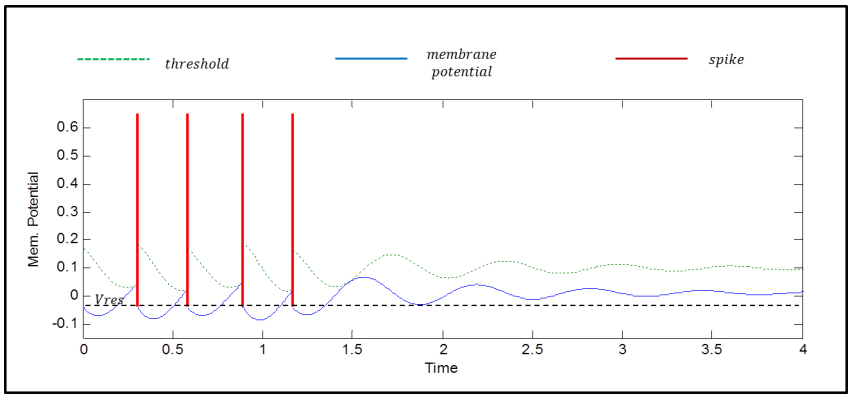

FIG. 47. **Case 3** time-voltage plot. The solution from Fig. 46 creates four spikes and then arrives to a resting state.

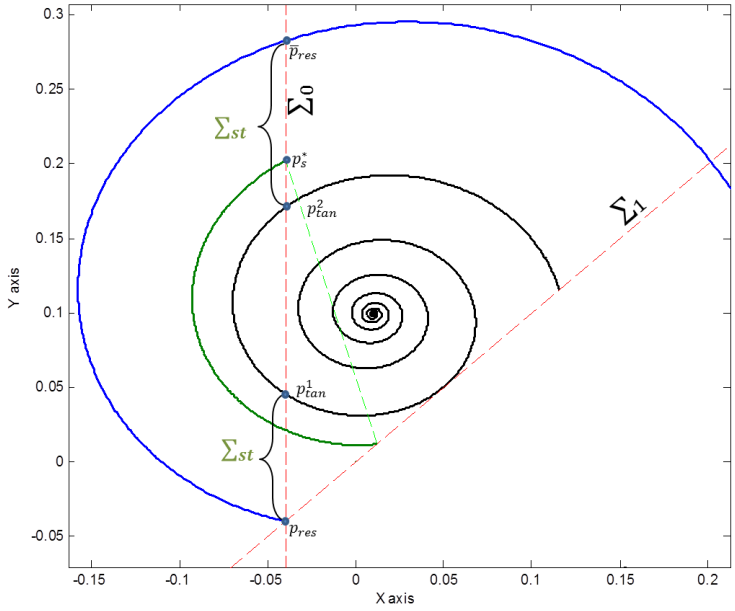

FIG. 48. **Case 3** The stable orbit defined on the second subdomain for the values: $V_{res} = -0.04$,$\Delta_y = 0.19$ and $y_s^* = 0.202$

Conclusion

We introduce a new spiking model consisting on a resonate and fire with a dynamic threshold ($MR\&F$). We study this model as a non-smooth (hybrid) dynamical system depending on two parameters V_{res} and Δ_y. A complete quantitative and qualitative analysis has been done to the modified resonate and fire model ($MR\&F$). We have introduced a one dimensional Poincaré map in the reset surface (curve). Fixed points, bifurcations and periodic orbits are found. A 3-dimensional diagram for the stable and unstable fixed points $y^*_{u,s}$ with respect to the parameters is computed as a tool to give an overall picture of the behavior of the fixed points and the asymptotic behavior of all the points. In particular, in the $MR\&F$ we find three different cases corresponding to three parameter regions, where the $MR\&F$ model shows different dynamics. Specially interesting is case 3 for which we find saddle-node bifurcations and sub-critical period doubling bifurcation where unstable period-2 points emerge from a stable fixed point. These bifurcations in the Poincaré map lead us to the corresponding bifurcation of periodic orbits of the system.

Something to remark for the $MR\&F$ is that we only consider a constant current input for the analysis. An advantage of this model is that, even with constant current input, for some values of the parameters V_{res} and Δ_y the model has periodic orbits. The parameters can be set in order to have only a stable periodic orbit, an unstable periodic orbit, both, or just solutions going directly to the focus which exists in the differential equation, whereas the Resonate and Fire model only shows a stable periodic orbit or solutions going directly to the focus. In other words, for our modified model we can have an unlimited sustained spiking, a limited sustained spiking or only sub-threshold oscillations depending on the initial conditions. The resonate and fire model could not show a limited sustained spiking, so the application of an stimulus current was required in order to achieve this. [2].

As this proposed model is derived from a previous one, possibly characteristics are inherited from the original model which were not studied here. For example the original system with periodic input has a preference to fire for pulses with inter-spikes intervals lower than the eigenperiod $T = 2\pi/\omega$ [2]. As we only did the analysis in the case of constant current input, then future works should include the analysis of a periodic stimulation, excitatory inputs or inhibitory inputs and consider also the inter-spikes intervals and how this affect the dynamics of the spiking. How solutions could intersect the different regions which are the basin of attraction of the stable periodic orbit and the focus respectively based on the spiking stimulus should be reviewed. A great variety of situations can be derived when the input is time dependent. This future analysis will contribute highly to confirm the validity of this model as a suitable resonator for modeling class 2 excitability.

This new proposed model resulted to have interesting characteristics that we do not doubt can have utilities in the neuroscience field. The $MR\&F$ showed to have damped oscillations, stable and unstable periodic orbits and bistability between fixed points and stable periodic orbits, which are common characteristic required for modeling a neuron while including threshold variability.

Appendix A
Programming Codes

In this chapter we are going to present the most important programs we compute to make the simulations. However, there are some modifications with the proposal to make different graphics and to see quantitative results that we have available.

1. Computation of $V - \theta$ Fourier Series

1.1. Main Program *FourierValues.m*.

```
fprintf('Welcome to Coeff Calc for Part Sols for the V-Theta Model\n');
fprintf('We will calculate the coefficients values for a Vp\n');
a_p=0.08; %Parameter a (ecuacion f(V)) fixed
b_p=4.9;  %Parameter b  (ecuacion f(V)) fixed
c_p=0.53;  %Parameter c (ecuacion f(V)) fixed
Vr_p=0.1;  %Parameter de Vr fixed
tao_p=2;   %Tao Constant
fprintf('The next parameters are going to be used\n');
fprintf('Vr=%.2f, a=%.2f, b=%.2f, c=%.2f, tao=%.2f ...
    \n',Vr_p,a_p,b_p,c_p,tao_p);
A = input('Specify a Value for A: ');
omega = input('Specify an omega: ');
T=1/omega;
fprintf('The period is %.4f \n', T);
fprintf('The omega is %.4f \n',omega)
n = input('Specify an index limit n: ');

[Vp,a_0,a_k,b_k,index]=Vp_sol(Vr_p,A,n,omega,1);

t=index*0.01;

figure(1);
plot(0.01:0.01:t,Vp);
xlim([0 T]) %Limits the wave to just one period
hold on;

elemVp=T*100; %How many elements has Vp in the range of a period
n_k = input('Specify an index limit n: ');

```

```
[fa0,fak,fbk]=areas(Vp,elemVp,a_p,b_p,c_p,tao_p,n_k);

Thp=Theta_sol(fa0,fak,fbk,tao_p,n_k,elemVp);
figure(2);
plot(0.01:0.01:T,Thp);
hold on;

figure(3);
plot(Vp(1:(T*100)),Thp(1:(T*100)));
line([-1 1], [-1 1],'Color','r','LineStyle','--'); %line([x1 x2], ...
    [y1 y2])<-----Sintaxis of Line
axis([-0.2 0.8 -0.2 0.8])
hold on;
```

1.2. Subprogram *Theta_sol.m*.

```
function [Thp]=Theta_sol(fa0,fak,fbk,tao_p,n_k,elemVp)

%First coefficient for the particular answer of theta
ta0=fa0*tao_p;  %Asignado
tak=zeros(1,n_k+1);   %Add zeros
tbk=zeros(1,n_k+1);   %Add zeros
Thp=zeros(1,elemVp);

Sum1=0;   %Var auxiliary
Sum2=0;    %Var auxiliary
tak(1)=0;    %Always will be zero
tbk(1)=0;    %Always will be zero

T=elemVp*0.01;

%Assign the new values for the coefficients of the solutions theta
for k=2:1:(n_k+1)
  auxval1=(fak(k)*tao_p-2*pi*(k-1)*fbk(k)*tao_p*tao_p/T);
  auxval2=(fbk(k)*tao_p+2*pi*(k-1)*fak(k)*tao_p*tao_p/T);
  tak(k)=((T*T)/(T*T+4*pi*pi*(k-1)*(k-1)*tao_p*tao_p))*auxval1;
  tbk(k)=((T*T)/(T*T+4*pi*pi*(k-1)*(k-1)*tao_p*tao_p))*auxval2;
end


%We evaluate for all the range of the period.
for time=1:1:elemVp
   t=time*0.01;
   Sum1=0;
   Sum2=0;
   for k=1:1:(n_k+1)
      Sum1=Sum1+tak(k)*cos(2*pi*(k-1)*t/T);
      Sum2=Sum2+tbk(k)*sin(2*pi*(k-1)*t/T);
   end
   Thp(time)=ta0+Sum1+Sum2;

end
```

1.3. Subprogram *Vp_sol.m*.

```
function [Vp,Va_0,Va_k,Vb_k,index]=Vp_sol(V_r,A,n,w,n_per)

%Generate the arrays with spaces for the indexes
T=1/w;
Va_k= zeros(1, n+1);
Vb_k= zeros(1, n+1);


Sum1=0;
Sum2=0;
%Generate the number of spaces for the range of time from 0 to T ...
    with a resolution of 0.01 in that interval
index=roundn((T*n_per/0.01),0);
Vp=zeros(1,index);

%Assign the values to the coefficientes for the indixed 0 and 1

Va_0=V_r+(A/pi); %Value for index a_0

Va_k(1)=0;  %Value for index 0 de ak
Vb_k(1)=0;  %Value for index 0 de bk

%Generate the following values for the next indixes n's

for k=2:1:(n+1)
 Va_k(k)=(1/(1+16*pi*(k-1)*(k-1)*w*w))*(2*A/pi)*(1/(1-4*(k-1)*(k-1)));
 Vb_k(k)=(1/(1+16*pi*(k-1)*(k-1)*w*w))*(4*pi*(k...
         -1)*w)*(2*A/pi)*(1/(1-4*(k-1)*(k-1)));
end

%Now we assign all the values of the indixes

for time=1:1:index
    t=time*0.01;
    Sum1=0;
    Sum2=0;
    for k=1:1:(n+1)
      Sum1=Sum1+Va_k(k)*cos(4*pi*w*(k-1)*t);
      Sum2=Sum2+Vb_k(k)*sin(4*pi*w*(k-1)*t);
    end

    Vp(time)=Va_0+(1/(1+4*pi*w*w))*(A/2)*sin(pi*t*2*w) ...
        +(1/(1+4*pi*w*w))*(-pi*w*A)*cos(pi*t*2*w)+Sum1+Sum2;
end
```

1.4. Subprogram *Areas.m*.

```
function [a0,ak,bk]=areas(Vp,elemVp,a_p,b_p,c_p,tao_p,n_k)

%Coeficientes que retornaremos, ponemos n_k+1 ya que consideramos ...
    el k=0
a0=0;
ak=zeros(1,n_k+1);
bk=zeros(1,n_k+1);

```

```
a0suma=0;
aksuma=0;
bksuma=0;

ak(1)=0;
bk(1)=0;


for td=2:1:(elemVp-1)
a0suma=a0suma+2*(a_p+exp(b_p*(Vp(td)-c_p)))/tao_p;
end

a0suma=a0suma+...
(a_p+exp(b_p*(Vp(1)-c_p)))/tao_p+...
(a_p+exp(b_p*(Vp(elemVp)-c_p)))/tao_p;

a0integral=a0suma*0.01/2; %fin de el calculo de integral de ...
    coeficiente a0


%1/2L osea 1/T, donde T es elemVp, pero se multiplica por 0.01 para ...
    pasarlo a segundos
a0=a0integral/(elemVp*0.01);



for k=2:1:(n_k+1)

    for td=2:1:(elemVp-1)

    ak(k)=ak(k)+...
        (2*(a_p+exp(b_p*(Vp(td)-c_p)))/tao_p)*cos(2*pi*(k-...
        1)*td*0.01/(elemVp*0.01));

    bk(k)=bk(k)+...
        (2*(a_p+exp(b_p*(Vp(td)-c_p)))/tao_p)*sin(2*pi*(k-...
        1)*td*0.01/(elemVp*0.01));

    end

    %Casi por finalizar el calculo por trapecio, falta sumar los lados
    %Calculo para coeficiente ak
    ak(k)=ak(k)+...
        ((a_p+exp(b_p*(Vp(1)-c_p)))/tao_p)*cos(2*pi*(k-...
        1)*td*0.01/(elemVp*0.01))+...
    ((a_p+exp(b_p*(Vp(elemVp)-c_p)))/tao_p)*cos(2*pi*(k-...
        1)*td*0.01/(elemVp*0.01));

    %Calculo para coeficiente bk
    bk(k)=bk(k)+...
        ((a_p+exp(b_p*(Vp(1)-c_p)))/tao_p)*sin(2*pi*(k-1...
        )*td*0.01/(elemVp*0.01))+...
    ((a_p+exp(b_p*(Vp(elemVp)-c_p)))/tao_p)*sin(2*pi*(k-...
        1)*td*0.01/(elemVp*0.01));

    %Sumando los lados para finalizar la integral
    ak(k)=ak(k)*0.01/2; %fin de el calculo de la integral de ak
    bk(k)=bk(k)*0.01/2; %fin de el calculo de la integral de bk
```

```
64
65      %Dividimos sobre 1/L la integral de ak
66      ak(k)=ak(k)/(elemVp*0.01/2);
67
68      %Dividimos sobre 1/L la integral de bk
69      bk(k)=bk(k)/(elemVp*0.01/2);
70
71  end
```

2. $V-\theta$ Spikes Simulation

This program simulates the spiking process of the integrate and fire with dynamic treshold.

2.1. Main Program *MultiNeuronRes.m*.

```
1   spikesarray=zeros(500,3);
2   counterarray=1;
3   %for thet=0.2:0.03:.5
4      % for vol=0:0.03:1
5
6           vol=0; %INITIAL CONDITIONS
7           thet=0.173; %INITIAL CONDITIONS
8
9
10           %Parameters of the model
11           Vreset=0;
12           theta_Δ=0.1;
13           ti=0; %Initial time
14
15           isfix=0; %Auxiliar variable used to detect fixed points.
16           nspikes=0;
17
18           %We generate a loop, that will continue to run until we ...
                 have not
19           %reached the fix point
20           while isfix==0
21           [val_f,time,tarray,yarray]=Neuron_Res(vol,thet,ti,1.0e-8);
22
23           if val_f(1,1)≥999
24               disp('ERROR increase time limit for integration');
25               break;
26           end
27
28           isdiag=roundn(val_f(1,1),-7)-roundn(val_f(1,2),-7);
29           switch (isdiag)
30               case 0
31                   nspikes=nspikes+1;
32                   spikesarray(nspikes,3)=nspikes;
33
34                 %CHART OF SPIKES IN A TIME SERIES
35                   figure(2); %Chart where we put the times series vs ...
                         voltage
```

```
36                  plot(tarray,yarray(:,1),'-',tarray,yarray(:,2),':');
37                  line([tarray(end) tarray(end)], [yarray(1,1) ...
                        0.65],'Color','r','LineStyle','-','LineWidth',2);
38                  axis([0 10 -0.15 0.7])
39                  hold on;
40                %END OF PLOTTING
41
42                  spikesarray(nspikes,1)=yarray(1,1);
43                  spikesarray(nspikes,2)=yarray(1,2);
44
45
46                  fprintf('Setting New Conditions after Spike \n\n');
47                  figure(1);
48                  line([val_f(1,1) Vreset], [val_f(1,2) ...
                        val_f(1,2)+theta_Δ],'Color','g','LineStyle','--');
49                  vol=Vreset;
50                  thet=val_f(1,2)+theta_Δ;
51                  ti=time;
52
53              otherwise
54                  disp('FINISH');
55                  figure(2);
56                  plot(tarray,yarray(:,1),'-',tarray,yarray(:,2),':');
57                  hold on;
58
59                  isfix=1;
60                  if(nspikes>0)
61                      counterarray=counterarray+1;
62                  end
63          end
64
65          end
66
67      %end
68
69  %end
```

2.2. Subprogram *Neuron_Res.m*.

```
1  %This function returns the value of the fixed point, or the value where
2  %the voltaje and the threshold are equals, if none of these happens in
3  %given time limit, then the program stops and send a message, and ...
       the time
4  %limit should be increased to a bigger value.
5  %Remark that in_x is VOLTAGE and that in_y is THETA
6
7  function [val_f,time,ta,ya]=Neuron_Res(in_x,in_y,i_time,reltol)
8  y0=[in_x;in_y]; %Initial conditions that come from MultiNeuronRes
9  a_p=0.08; %Parameter a (ecuacion f(V)) fixed
10 b_p=4.9;  %Parameter b  (ecuacion f(V)) fixed
11 c_p=0.53;  %Parameter c (ecuacion f(V)) fixed
12 Vr_p=0.1;  %Parameter de Vr fixed
13 I_p=0.3;   %Constant Current
14 tao_p=2;   %Tao Constant
15 fp=[Vr_p+I_p;a_p+exp(b_p*(Vr_p+I_p-c_p))];
16 %We put the conditions on the integrator
```

```
%who stop integrating when V-Theta=0
%these exceptions are on the gstop_res function
opts=odeset('events',@gstop_res,'reltol',reltol,'refine',9);

%Integrator of ODE that integrate from 0 to 10*Pi
%and return the values of the array of the integration,
%where ta contains all the points of the integration until tfinal
%or when the event happen, in case the event happen, then
%the variable ye and te are filled with the last values of ta and ya
[ta,ya,te,ye]=ode23(@(t,y,y0,a,b,c,Vr,I,tao)Vtheta(t,y,y0,...
              a_p,b_p,c_p,Vr_p,I_p,tao_p),[i_time ...
                  5*pi+i_time],y0,opts,y0);

%Here we put in val_f the last values of ya obtained, independently of
%an event or tfinal reached.
val_f=ya(end,:);
val_sf=ya(end-1,:);
time=ta(end);

if (roundn(val_f(1,1)-val_f(1,2),-5))==0
   fprintf('Threshold Reached - Neuron Spiked at %.5f seconds \n',time)

   elseif (roundn(val_f(1,1)-val_sf(1,1),-4)==0 && ...
       roundn(val_f(1,2)-val_sf(1,2),-4)==0)
       fprintf('Computation Finished-Approximation of 1x10-4 to the ...
           fix point (V,T) Fix point obtained %.6f %.6f Real Fix ...
           Point %.6f %.6f \n',val_f(1,1),val_f(1,2),fp(1),fp(2));
       fprintf('Total Computation time of %.5f seconds \n', time);

   else
       fprintf('Error! We need more time to reach a fix point or a ...
           Threshold %.5f %.5f %.6f ...
           %.6f',val_f(1,1),val_f(1,2),fp(1),fp(2));
       fprintf('Total Computation time of %.5f seconds \n', time);
       val_f=[999,999];
end
%tfinal=ta(end)
%yfinal=ye(end,1:2)

%Here we plot the solution curve obtained previously
figure(1);
plot(ya(:,1),ya(:,2),'-',0,0,'ro')

%we set the diagonal line when V=Theta, axis, title and labels
line([-1 1], [-1 1],'Color','r','LineStyle','--'); %line([x1 x2], ...
    [y1 y2])<-----Sintaxis of Line
axis([-1 2.0 -1 2.0])
hold off;
title('V-Theta Model');
ylabel('Theta');
%xlabel('Voltage');
m=sprintf('Voltage \n Parameters a:%.2f  b:%.2f  c:%.2f  Vr:%.2f  ...
    I:%.2f  Tao:%.2f',a_p,b_p,c_p,Vr_p,I_p,tao_p);

xlabel(m);

%If we wanna plot many solutions in the same chart.
hold on;
```

2.3. Subprogram *Vtheta.m*.

```
%Equations of V-Theta System
function ydot = Vtheta(t,y,y0,a,b,c,Vr,I,tao)
ydot=[-y(1)+Vr+I;-(y(2)-(a+exp(b*(y(1)-c))))/tao];
```

2.4. Subprogram *gstop_res.m*.

```
function [val,isterm,dir] = gstop_res(t,y,y0)

val = [y(1)-y(2)];

isterm = [1]; %Terminate function when the value get zero
dir = 0; %Direction where we reach the zero dont matter
```

3. $MR\&F$ Spikes Simulation

This program simulates the spiking process of the resonate and fire with dynamic treshold.

3.1. Main Program *MultiNeuronRes.m*.

```
spikesarray=zeros(500,3);
counterarray=1;

%for YCor=0:0.1:1
    %for XCor=0:0.03:1


  XCor=-.05; %INITIAL CONDITIONS
  YCor=.24; %INITIAL CONDITIONS

        %Parameters of the model
        Vreset=-0.05;
        Y_Δ=.03;
        ti=0; %Initial time

        isfix=0; %Auxiliar variable used to detect fixed points.
        nspikes=0;



        %We generate a loop, that will continue to run until we ...
            have not
        %reached the fix point
        while isfix==0
        [val_f,time,tarray,yarray]=Neuron_Res(XCor,YCor,ti,1.0e-8,Vreset);

        if val_f(1,1)≥999
            disp('ERROR increase time limit for integration');
```

```
28              break;
29          end
30
31          isdiag=roundn(val_f(1,1),-7)-roundn(val_f(1,2),-7);
32          switch (isdiag)
33              case 0
34                  nspikes=nspikes+1;
35                  spikesarray(nspikes,3)=nspikes;
36
37                  %SPIKES CHART IN A TIME SERIES
38                  figure(2);
39                  plot(tarray,yarray(:,1),'-',tarray,yarray(:,2),':');
40                  line([tarray(end) tarray(end)], [yarray(1,1) ...
                        0.65],'Color','r','LineStyle','-','LineWidth',2);
41                  axis([0 10 -0.15 0.7])
42                  hold on;
43                  %END OF THE CHART
44
45
46                  spikesarray(nspikes,1)=yarray(1,1);
47                  spikesarray(nspikes,2)=yarray(1,2);
48
49                  fprintf('Setting New Conditions after Spike \n\n');
50                  figure(1);
51                  line([val_f(1,1) Vreset], [val_f(1,2) ...
                        val_f(1,2)+Y_Δ],'Color','cyan','LineStyle','--','LineWidth',1);
52                  XCor=Vreset;
53                  YCor=val_f(1,2)+Y_Δ;
54                  ti=time;
55
56
57
58              otherwise
59                  disp('FINISH');
60                 figure(2);
61                  plot(tarray,yarray(:,1),'-',tarray,yarray(:,2),':');
62                  hold on;
63
64                  isfix=1;
65                  if(nspikes>0)
66                      counterarray=counterarray+1;
67                  end
68          end
69
70          end
71
72      %end
73
74  %end
```

3.2. Subprogram *Neuron_Res.m*.

```
1  %This function returns the value of the fixed point, or the value where
2  %the voltaje and the threshold are equals, if none of these happens in
3  %given time limit, then the program stops and send a message, and ...
       the time
```

```
%limit should be increased to a bigger value.

function [val_f,time,ta,ya]=Neuron_Res(in_x,in_y,i_time,reltol,Vreset)
y0=[in_x;in_y]; %Initial conditions that come from MultiNeuronRes

b_p=-1;
w_p=10;
I_p=1;

fp=[-(b_p*I_p)/(b_p^2+w_p^2);(w_p*I_p)/(b_p^2+w_p^2)];
%We put the conditions on the integrator
%who stop integrating when x-y=0
%these exceptions are on the gstop_res function
opts=odeset('events',@gstop_res,'reltol',reltol,'refine',9);

%Integrator of ODE that integrate from 0 to 3*Pi
%and return the values of the array of the integration,
%where ta contains all the points of the integration until tfinal
%or when the event happen, in case the event happen, then
%the variable ye and te are filled with the last values of ta and ya
[ta,ya,te,ye]=ode23(@(t,y,y0,b,w,I)Resonatefire(t,y,y0,b_p,w_p,I_p),[i_time ...
    3*pi+i_time],y0,opts,y0);

%Here we put in val_f the last values of y obtained, independently of
%an event or tfinal reached.
val_f=ya(end,:);
val_sf=ya(end-1,:);
time=ta(end);

if (roundn(val_f(1,1)-val_f(1,2),-5))==0
   fprintf('Threshold Reached - Neuron Spiked at %.5f seconds \n',time)

   elseif (roundn(val_f(1,1)-val_sf(1,1),-4)==0 && ...
       roundn(val_f(1,2)-val_sf(1,2),-4)==0)
       fprintf('Computation Finished-Approximation of 1x10-4 to the ...
           fix point (V,T) Fix point obtained %.6f %.6f Real Fix ...
           Point %.6f %.6f \n',val_f(1,1),val_f(1,2),fp(1),fp(2));
       fprintf('Total Computation time of %.5f seconds \n', time);

   else
       fprintf('Error! We need more time to reach a fix point or a ...
           Threshold %.5f %.5f %.6f ...
           %.6f',val_f(1,1),val_f(1,2),fp(1),fp(2));
       fprintf('Total Computation time of %.5f seconds \n', time);
       val_f=[999,999];
end
%tfinal=ta(end)
%yfinal=ye(end,1:2)

%Here we plot the solution curve obtained previously
figure(1);
plot(ya(:,1),ya(:,2),'-','Linewidth',1,'Color','b');

%we set the diagonal line when V=Theta, axis, title and labels
line([-3 2], [-3 2],'Color','r','LineStyle','--'); %line([x1 x2], ...
    [y1 y2])<-----Sintaxis of Line
line([Vreset Vreset], [-2 2],'Color','r','LineStyle','--');
axis([-2 2.0 -2 2.0])
```

```
55 hold off;
56 title('Resonate and Fire');
57 ylabel('Y axis');
58 xlabel('Voltage');
59 m=sprintf('X axis \n Parameters b:%.2f  w:%.2f  I:%.2f',b_p,w_p,I_p);
60
61 xlabel(m);
62
63 %If we wanna plot many solutions in the same chart.
64 hold on;
```

3.3. Subprogram *Resonatefire.m*.

```
1 %Equations of MRF Model
2 function ydot = Resonatefire(t,y,y0,b,w,I)
3 ydot=[b*y(1)-w*y(2)+I;w*y(1)+b*y(2)];
```

3.4. Subprogram *gstop_res.m*.

```
1 function [val,isterm,dir] = gstop_res(t,y,y0)
2
3 val = [y(1)-y(2)];
4
5 isterm = [1]; %Terminate function when the value get zero
6 dir = 0; %Direction where we reach the zero dont matter
```

4. Computation of Maps for the $MR\&F$ Model

The additional program calculate the maps for the case 3 of the $MR\&F$ Model. In the next web address the program for the maps for case 1 and 2 and additional auxiliary programs (fix points, bifurcations) can be found: https://www.dropbox.com/sh/ra76asnoepfnoap/XJtacchlH1.

4.1. Main Program *Maps.m*.

```
1 Vreset=-0.05; \PARAMETER
2 Deltay=0.13;   \PARAMETER
3 [y1_tan, yvalues1]=Plot(Vreset,Deltay);
4 [y2_tan, yvalues2]=Plot2R(y1_tan,Vreset,Deltay);
5 line([-0.1 1], [-0.1 1],'Color','r','LineStyle','--');
```

4.2. Subprogram *Plot.m*.

```
1 function [y1_tan,yvalues1]=Plot(Vres,Δy)
2 %Parameters for the model
3 b_p=-1; %b
4 w_p=10;  %omega
```

```
I_p=1;  %input
%Vres=-0.05; %Vreset
%Δy=0.01; %Deltay
Ptan=I_p/(2*w_p); %Tangent point
syms ti;

%Costant Terms repetitive terms
ta_1=b_p^2+w_p^2;
pf_x=-b_p*I_p/(ta_1);
pf_y=w_p*I_p/(ta_1);
ti_inter=0;
seed=0;
tiempo=0;
contador=0;
%**********************************************
%Calculate the intersection of curve with Vres.
%**********************************************

%Formula for getting radio0 & theta0 from the tangent point.
%this calculation is for plotting the drawing of the solution that ...
    defines
%the spikings, and for setting the equation that will be solved for ...
    finding
%y^1_tan

[r_0,theta_0]=polares(Ptan,Ptan,pf_x,pf_y,b_p,w_p,I_p,ta_1);

%This loop calculates the solution that defines the boundaries of
%solutions that go to the equilibrium point, this loop start from the
%tangent point and then go backwards in time until a given limit ...
    stablished
for tiempo=.001:.001:3
    array=tiempo*1000;
    array=roundn(array,0);
    x(array)=pf_x+r_0*exp(b_p*(-tiempo))*cos(theta_0+w_p*(-tiempo));
    y(array)=pf_y+r_0*exp(b_p*(-tiempo))*sin(theta_0+w_p*(-tiempo));
    if x(array)<Vres && x(array-5)>Vres
        contador=contador+1;
        if contador==1;
           seed=tiempo;
        end
    end
end
%figure(2);
%plot(x,y);
%hold on;
%line([Vres Vres], [-2 2],'Color','r','LineStyle','--');
%line([-3 2], [-3 2],'Color','r','LineStyle','--'); %line([x1 x2], ...
    [y1 y2])<-----Sintaxis of Line


%Calculation of Time for intersection the Vres from Ptan
%Equation of X=Vres we find the zeros for a given ti (time ...
    intersection)
%the more near to zero but positive
while roundn(ti_inter,-8)≥0
options=optimset('Display','iter',...
```

```
'Algorithm','trust-region-dogleg','TolFun',1.000000e-7);   % Option ...
    to display output

pointf=@(ti)-Vres+pf_x+r_0*exp(b_p*(-ti))*cos(theta_0+w_p*(-ti));
ti_inter=fsolve(pointf,seed,options);
    if roundn(ti_inter,-8)==0 || roundn(ti_inter,-8)<0
        seed=seed+.001;
        ti_inter=0;
    else roundn(ti_inter,-8)>0;
            break;
    end
end

%Once we know tf=ti_inter, we calculate y(tf) that is the value of ...
    y1_tan,
%we do that with the equations setting them from t --->  -t
P1_x=pf_x+r_0*exp(b_p*(-ti_inter))*cos(theta_0+w_p*(-ti_inter));
P1_y=pf_y+r_0*exp(b_p*(-ti_inter))*sin(theta_0+w_p*(-ti_inter));

y1_tan=P1_y;

%The number of positions are calculated
positions=roundn((P1_y-Vres-.0001)/.0001,0);
yvalues=zeros(positions,5);

%With this loop we calculate with the initial conditions from ...
    Sigma_0 that
%goes from V_res,y^{1}_tan to
%the solution where x(tf)=y(tf) with a tf>0 and more nearer to ...
    zero, then
%y(tf) is calculated and stored in an array, where in another array we
%store the image of y(tf)+\Δ_y, finally is plotted y(tf) vs
%y(tf)+\Δ_y for all the y_0 from the initial conditions that goes ...
    from y^{1}_tan to V_res

for iter=1:1:positions
%We begin with the initial condition from Vres (.0001 little more ...
    bigger
%than Vres),then the positions are increased for every loop by a ...
    quantity
%of .0001
y_0=Vres+iter*0.0001;

%we define the variable tf (the one we wanna find)
syms ti
%we write the equation x(tf)=y(tf)
f=@(ti)I_p*(w_p+b_p)/(b_p^2+w_p^2)+exp(b_p*ti)*((y_0-Vres-...
       I_p*(b_p+w_p)/(b_p^2+w_p^2))*cos(w_p*ti)+...
     ((I_p*(-w_p+b_p)/(b_p^2+w_p^2))+y_0+Vres)*sin(w_p*ti));
%we solve it looking for the time more nearer to zero (but positive)

seed=0;
while roundn(tiempo,-8)≥0
options=optimset('Display','iter','Algorithm','trust-region-dogleg',...
      'TolFun',1.000000e-7);   % Option to display output

f=@(ti)I_p*(w_p+b_p)/(b_p^2+w_p^2)+...
      exp(b_p*ti)*((y_0-Vres-I_p*(b_p+w_p)/(b_p^2+...
```

```
        w_p^2))*cos(w_p*ti)+...
             ((I_p*(-w_p+b_p)/(b_p^2+w_p^2))+y_0+Vres)*sin(w_p*ti));

tiempo=fsolve(f,seed,options);
    if roundn(tiempo,-8)==0 || roundn(tiempo,-8)<0
        seed=seed+.01;
        tiempo=0;
    else roundn(tiempo,-8)>0;
            break;
    end
end

%the time is stored in a variable called ti
ti=tiempo;

%The variable y(tf) is calculated with the previous time found, and ...
    y(tf) should be equal to x(tf)
y_tf=(w_p.*I_p)/(b_p^2+w_p^2)+exp(b_p*ti)*((y_0-w_p*I_p/(b_p^2+...
      w_p^2))*cos(w_p*ti)+(Vres+b_p*I_p/(w_p^2+b_p^2))*sin(w_p*ti));
x_tf=-(b_p.*I_p)/(b_p^2+w_p^2)+exp(b_p*ti)*(-(y_0-w_p*I_p/(b_p^2+...
      w_p^2))*sin(w_p*ti)+(Vres+b_p*I_p/(w_p^2+b_p^2))*cos(w_p*ti));

%Finally we store an array, the initial condition, the value for x(tf),
%y(tf),tf and y(tf)+\Δ_y
yvalues(iter,1)=y_0;
yvalues(iter,2)=x_tf;
yvalues(iter,3)=y_tf;
yvalues(iter,4)=tiempo;
yvalues(iter,5)=y_tf+Δy;

end

%The information is plotted in a new window
figure(1);
hold on;

%Finally we plot in a chart the values for y_0 and y_0+\Δ_y
plot(yvalues(:,1),yvalues(:,5));
xlabel('Y_0');
ylabel('Y_0 + \Delta_y');
yvalues1=yvalues;
%line([-0.1 1], [-0.1 1],'Color','r','LineStyle','--');
```

4.3. Subprogram *Plot2R.m*.

```
function [y2_tan,yvalues2]=Plot2R(y1_tan,Vres,Δy)
%Parameters for the model
b_p=-1; %b
w_p=10;  %omega
I_p=1;  %input
%Vres=-0.05; %Vreset
%Δy=0.01; %Deltay
Ptan=I_p/(2*w_p); %Tangent point
tiempo=0;
syms ti;
syms ti2;
```

```
12
13 %Costant Terms repetitive terms
14 ta_1=b_p^2+w_p^2;
15 pf_x=-b_p*I_p/(ta_1);
16 pf_y=w_p*I_p/(ta_1);
17 ti_inter=0;
18 ti_inter2=0;
19 seed=0;
20
21 %*********************************************
22 %Calculate the intersection of curve with Vres.
23 %*********************************************
24
25 %Formula for getting radio0 & theta0 (initial conditions) from the ...
       (Vres,y1_tan) this will be
26 %used for setting the initial conditions for the equations that ...
       once solved
27 %will give us y2_tan
28 [r_0,theta_0]=polares(Vres,y1_tan,pf_x,pf_y,b_p,w_p,I_p,ta_1);
29
30 %for tiempo=.001:.001:3
31  %   array=tiempo*1000;
32   %  array=roundn(array,0);
33   %  x(array)=pf_x+r_0*exp(b_p*(-tiempo))*cos(theta_0+w_p*(-tiempo));
34   %  y(array)=pf_y+r_0*exp(b_p*(-tiempo))*sin(theta_0+w_p*(-tiempo));
35 %end
36 %hold on;
37 %figure(1);
38 %plot(x,y);
39
40
41 %Formula for getting the values for x(tf)=y(tf)=Vres this will be ...
       used for
42 %the equations
43 [r_1,theta_1]=polares(Vres,Vres,pf_x,pf_y,b_p,w_p,I_p,ta_1);
44
45 %THE NEXT LOOP IS FOR FINDING THE TIME ELAPSED FROM (VRES,Y1_TAN) TO
46 %(VRES,Y2_TAN) (EQUATION IS BACKWARD IN TIME)
47 while roundn(ti_inter,-8)≥0
48 pointf=@(ti)-Vres+pf_x+r_0*exp(b_p*(-ti))*cos(theta_0+w_p*(-ti));
49 ti_inter=fzero(pointf,seed);
50     if roundn(ti_inter,-8)==0 || roundn(ti_inter,-8)<0
51         seed=seed+.01;
52         ti_inter=0;
53     else roundn(ti_inter,-8)>0;
54             break;
55     end
56 end
57
58 %THE NEXT LOOP IS FOR FINDING THE TIME ELAPSED FROM (VRES,VRES) TO
59 %(VRES,Y_RES) (EQUATION IS BACKWARD IN TIME)
60
61 seed=0;
62 while roundn(ti_inter2,-8)≥0
63 pointf1=@(ti2)-Vres+pf_x+r_1*exp(b_p*(-ti2))*cos(theta_1+w_p*(-ti2));
64 ti_inter2=fzero(pointf1,seed);
65     if roundn(ti_inter2,-8)==0 || roundn(ti_inter2,-8)<0
66         seed=seed+.01;
```

```
67          ti_inter2=0;
68      else roundn(ti_inter2,-8)>0;
69              break;
70      end
71  end
72
73  %WE NOW CALCULATE THE POINT Y2_TAN (FOR BOTH X AND Y)
74  P1_x=pf_x+r_0*exp(b_p*(-ti_inter))*cos(theta_0+w_p*(-ti_inter));
75  P1_y=pf_y+r_0*exp(b_p*(-ti_inter))*sin(theta_0+w_p*(-ti_inter));
76
77
78  %WE NOW CALCULATE THE POINT Y_RES (FOR BOTH X AND Y)
79  Pup1_x=pf_x+r_1*exp(b_p*(-ti_inter2))*cos(theta_1+w_p*(-ti_inter2));
80  Pup1_y=pf_y+r_1*exp(b_p*(-ti_inter2))*sin(theta_1+w_p*(-ti_inter2));
81
82  y2_tan=P1_y;
83
84  %Calculate distance from Y2_TAN to YRES
85  positions=roundn((Pup1_y-P1_y-.0001)/.0001,0);
86  yvalues=zeros(positions,5);
87
88  %t=0.1:0.001:20;
89  %x=-(b_p.*I_p)/(b_p^2+w_p^2)+r_0.*exp(b_p.*t).*cos(theta_0+w_p.*t);
90  %y=(w_p.*I_p)/(b_p^2+w_p^2)+r_0.*exp(b_p.*t).*sin(theta_0+w_p.*t);
91  %y_0=0.1931;
92
93  for iter=1:1:positions
94  %We begin with the initial condition from (VRES,Y2_TAN) (.0001 ...
        little more bigger
95  %than Vres),then the positions are increased for every loop by a ...
        quantity
96  %of .0001 UNTIL ARRIVE TO (VRES,Y_RES)
97  y0=y2_tan+iter*0.0001;
98
99  %we define the variable tf (the one we wanna find)
100 syms ti
101
102 %we solve it looking for the time more nearer to zero (but positive)
103 seed=0;
104 while roundn(tiempo,-8)≥0
105 f=@(ti)I_p*(w_p+b_p)/(b_p^2+w_p^2)+exp(b_p*ti)*((y0-Vres-...
106     I_p*(b_p+w_p)/(b_p^2+w_p^2))*cos(w_p*ti)+...
107          (+(I_p*(-w_p+b_p)/(b_p^2+w_p^2))+y0+Vres)*sin(w_p*ti));
108
109 tiempo=fzero(f,seed);
110     if roundn(tiempo,-8)==0 || roundn(tiempo,-8)<0
111         seed=seed+.005;
112         tiempo=0;
113     else roundn(tiempo,-8)>0;
114             break;
115     end
116 end
117 %the time is stored in a variable called ti
118 ti=tiempo;
119
120 y_tf=(w_p.*I_p)/(b_p^2+w_p^2)+exp(b_p*ti)*((y0-w_p*I_p/(b_p^2+...
121       w_p^2))*cos(w_p*ti)+(Vres+b_p*I_p/(w_p^2+b_p^2))*sin(w_p*ti));
122 x_tf=-(b_p.*I_p)/(b_p^2+w_p^2)+exp(b_p*ti)*(-(y0-w_p*I_p/(b_p^2+...
```

```
123         w_p^2))*sin(w_p*ti)+(Vres+b_p*I_p/(w_p^2+b_p^2))*cos(w_p*ti));
124
125 yvalues(iter,1)=y0;
126 yvalues(iter,2)=x_tf;
127 yvalues(iter,3)=y_tf;
128 yvalues(iter,4)=tiempo;
129 yvalues(iter,5)=y_tf+Δy;
130
131 end
132
133
134 figure(1);
135 hold on;
136 plot(yvalues(:,1),yvalues(:,5));
137 xlabel('Y0');
138 ylabel('Y0 + \Delta_y');
139 yvalues2=yvalues;
140 %line([-0.1 1], [-0.1 1],'Color','r','LineStyle','--');
```

4.4. Subprogram *polares.m*.

```
1  function [r,theta]=polares(x,y,pf_x,pf_y,b_p,w_p,I_p,ta_1)
2  if (x-pf_x)>0 && (y-pf_y)≥0;
3      theta=atan((y*ta_1-(w_p*I_p))/(x*ta_1+(b_p*I_p)));
4  elseif (x-pf_x)>0 && (y-pf_y)<0;
5      theta=atan((y*ta_1-(w_p*I_p))/(x*ta_1+(b_p*I_p)))+2*pi;
6  elseif (x-pf_x)<0 && (y-pf_y)<0;
7      theta=atan((y*ta_1-(w_p*I_p))/(x*ta_1+(b_p*I_p)))+pi;
8  elseif (x-pf_x)==0 && (y-pf_y)>0;
9      theta=pi/2;
10 elseif(x-pf_x)==0 && (y-pf_y)<0;
11       theta=3*pi/2;
12 end
13
14 r=sqrt((x+b_p*I_p/ta_1)^2+(y-pf_y)^2);
```

References

[1] E. M. Izhikevich. *Dynamical systems in neuroscience: the geometry of excitability and bursting*. Computational Neuroscience. MIT Press, Cambridge, MA, 2007.

[2] Keener JP, Hoppensteadt FC, Rinzel J. *Integrate and fire models of nerve membrane response to oscillatory input*. SIAM J Appl Math 41:503-17, 1987.

[3] E. M. Izhikevich. *Resonate-and-fire neurons*. Neural Networks, 14:883-8943, 2001.

[4] Meng X, Huguet G, Rinzel J. *Type III excitability, slope sensitivity and coincidence detection*. Discrete and continuous dynamical systems. Series A, Vol. 32, Núm. 8, (Pág. 2729-2757), 2012.

[5] Alligood, Kathleen T, Tim D, Yorke JA. *Chaos. An introduction to dynamical systems*. Springer Verlag. ISBN 0-387-94677-2, 2000.

[6] Arrowsmith DK, Place CM. *An Introduction to Dynamical Systems*. Cambridge University Press. ISBN 13:978-0521316507, 1990.

[7] Coddington W, Levinson N, *Theory of Ordinary Differential Equations*. McGraw-Hill, New York, 1955.