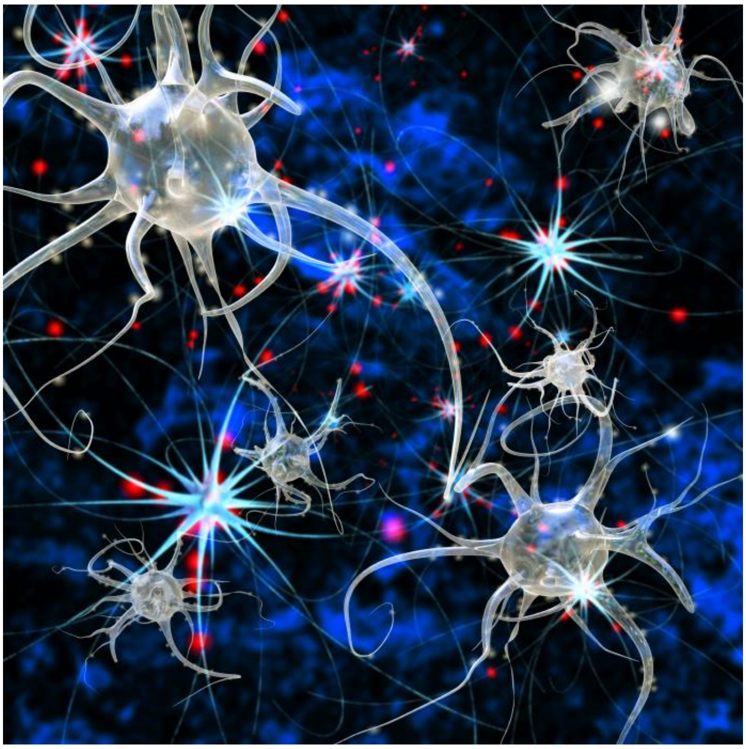


EXCITABLE SYTEM

Fitzhugh-Nagumo model

Modulation of the spike that is generated by a squid gian axon

Vega Vázquez Ardid

Biological Dynamic System Analysis

PHASE DIAGRAM

What parameters are constant along the different scenarios?
a,b and tau
What are varying?
I, constant stimulus current
How many scenarios are defined?
Three
What is the time interval and how many points are depicted?
The time interval is 0-200ms
And 1500 points are depicted

FITZHUGH-NAGUMO MODEL

Is a linear or a non-linear model?
This behavior is typical for spike generations (a short, non linear elevation of membrane voltage v, diminished over time by a slower, linear recovery variable w) in a neuron after stimulation by an external input current
The equations for this dynamical system:

$$\dot{v} = v - \frac{v^3}{3} - w + RI_{\text{ext}}$$

$$\tau\dot{w} = v + a - bw.$$

As we can see the fitzhugh-nagumo model is a simplified 2D version of hodgkin-huxley model
The motivation for the fitzhugh-naguno model was to isolate conceptually the essentially mathematical properties of excitation and propagation from the electrochemical properties of sodium and potassium ion flow.
The model consists of:

- A voltage-like variable having a cubic non-linearity that allows regenerative self-excitation via a positive feedback
- A recovery variable having a linear dynamics that provides a slower negative feedback

Thus we can conclude, the stimulation is non-linear but the recovery is linear.

What is the order of the model?
The stimulation differential equation is order 3(non-linear):

$$\dot{V} \quad = \quad V - V^3/3 - W + I$$

The recovery linear ODE order is 1 (linear):

$$\dot{W} \quad = \quad 0.08(V + 0.7 - 0.8W)$$

Justify each answer.
After programming the code you should see the simulations of the three scenarios. In each simulation, you should see three trajectories corresponding to three different initial conditions.
Show the three pictures and give a brief explanation of them
To explain the dynamics of this model we need to repeatedly change various inputs without changing the underlying model
The resting point p is stable, if a stimulus consisting of an instantaneous shock is applied to the system, the phase point jumps horizontally along the dotted line for a distance Δx proportional to the amplitude of the shock. After the shock is enough the phase point will shift to the right till absolutely refractory period, downward to relatively refractory period and back to p, this circuit represents a complete action potential:

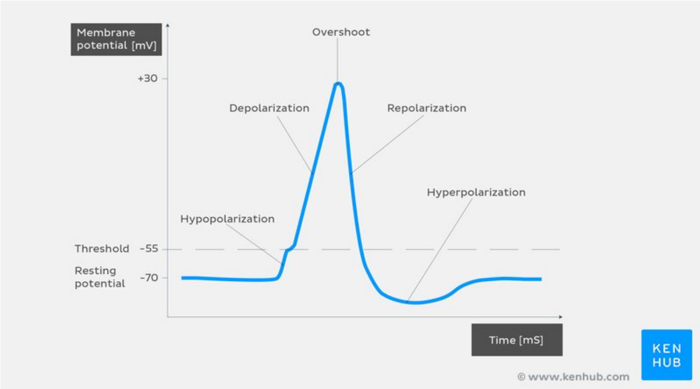
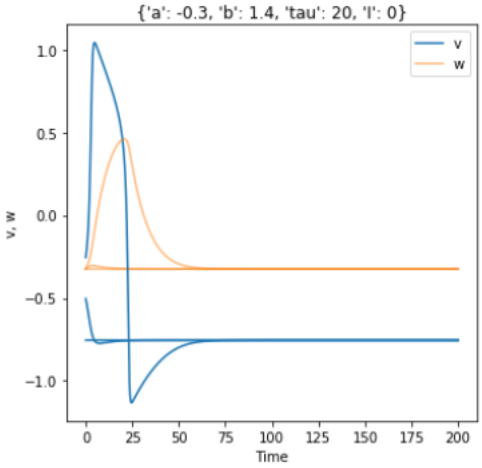
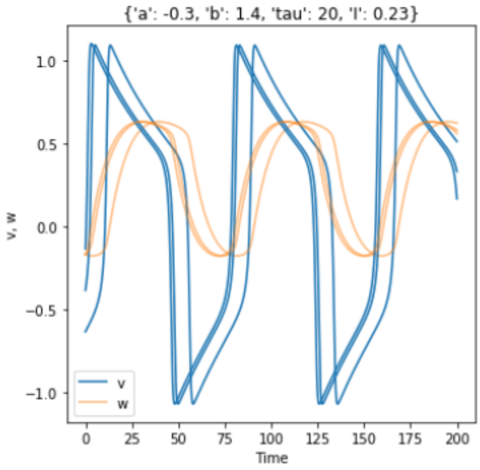


Figure 1



Parameters a and b and time scale constant
Stimulus current null
If the current is too small, no impulse results and the phase point will return more directly to P. Figure 2
Hypopolarization in Figure 1

Figure 2



If the current is larger than before V is moved past -0,64 v (threshold) then action potential will occur.
Moreover, if we stimulate at t= 0 with a sustained current step, the system will remain stable if the real parts of both eigenvalues are negative. Figure 3
Depolarization, Overshot and repolarization in Figure 1

Figure 3

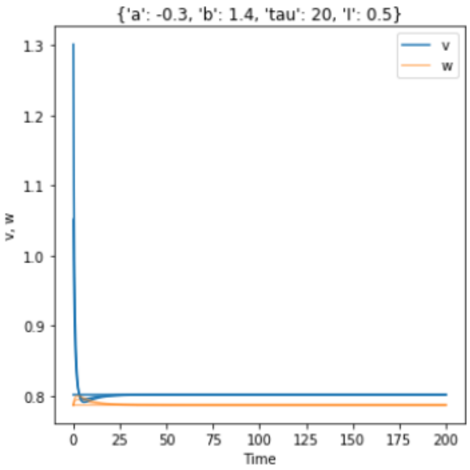


Figure 4

Once the real part of one of the eigenvalues becomes 0 and the other positive, the smallest perturbations will become amplified and diverge away from the equilibrium. Figure 4

Hyperpolarization in figure 1

ISOCLINES

NULL-CLINES

Write the two equations as w = F(v).

What kind of functions are the null-clines? Program the function 8 to plot the null-clines

Show the null-clines for the three scenarios in your document.

$$\frac{dv}{dt} = 0$$

$$v - v^3 - w + I = 0$$
$$\boldsymbol{v - v^3 + I = w}$$

$$\frac{dw}{dt} = 0$$

$$\boldsymbol{\frac{v - a}{b} = w}$$

v = np.linspace(vmin,vmax,100) – generating an array form by 100 equidistance numbers between maximum voltage and minimum voltage –

w1 = (v - v**3 + I) – null-cline for the first differential equation–

w2 = ((v - a)/b) – null-cline for the second differential equation –

This kind of bifurcation is clearly an imperfect bifurcation of the form

$\frac{dx}{dt} = h + rx - x^3$, where h is the imperfection parameter that breaks the simmetry

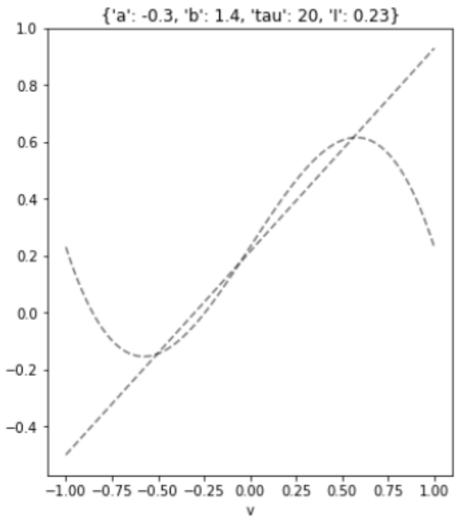


Figure 5

Whenever the stimulus is enough to surpass the threshold the phase point shifts to the right and w meets v giving rise to three fixed points: depolarization, overshoot and repolarization from left to right
Three intersection points.
Figure 5

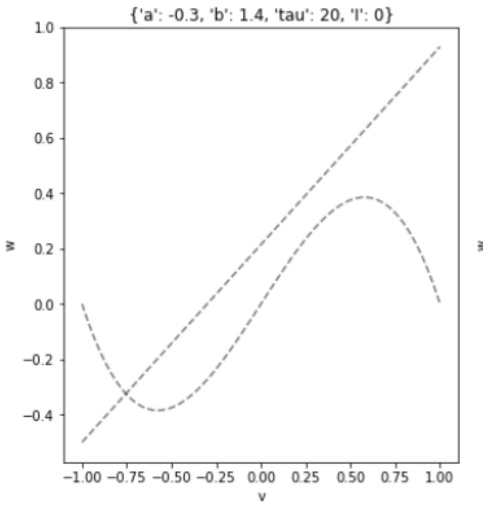


Figure 6

As we can see, the external current here is very small so the system will immediately return to the rest following a tight trajectory around the equilibrium points

$$\boldsymbol{v - v^3 + I = w}$$

$$\boldsymbol{\frac{v-a}{b} = w}$$

The hypopolarization is produced because the current is not enough to surpass the threshold, thus not producing any action potential.
One fixed point.
Figure 6

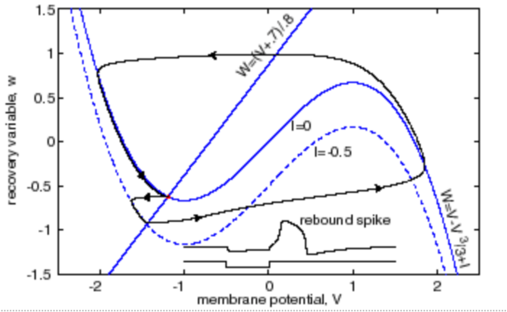


Figure 7

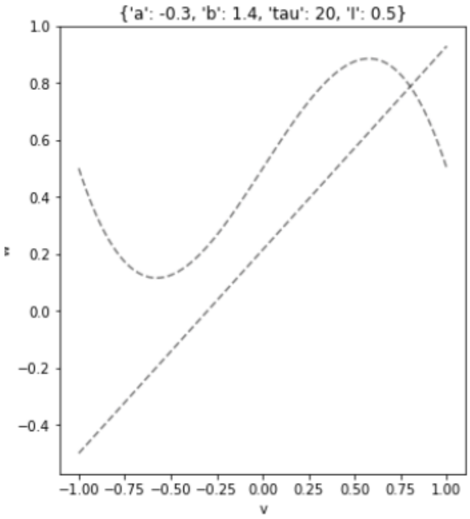


Figure 8

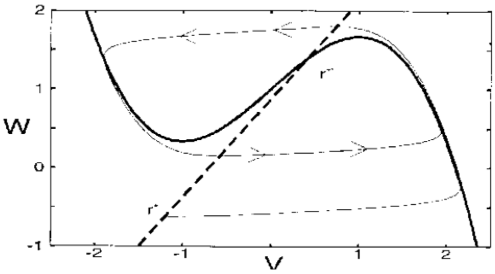


Figure 9

If the external current is large enough to surpass the threshold, the system is shifted to the right, moving away from the V nullcline
It explains the dynamical mechanism of spike accommodation in HH-type models.
I must increase slowly thus the cell remains quiescent and the equilibrium shifts to the right, without firing spikes
As long as the real part of the eigenvalues is negative, the fixed point is stable, but once it becomes positive, even smallest perturbations will become amplified and diverge away from the equilibrium
When stimulation is increased abruptly, even by a smaller amount, the trajectory could not go directly to the new resting state so it fires a transient spike, very similar to the anodal break excitation.
One intersection point .
Figure 9

FLOW

Compute the flow and give it the expression in your work. Look for the piece of code similar to the following one and add the sentence needed to run it:
Show the flow of the three scenarios on your document.

x = np.linspace(xrange[0], xrange[1], steps), array between 0-1, steps points

y = np.linspace(yrange[0], yrange[1], steps), array between 0-1, step points

X,Y = np.meshgrid(x,y), to create a rectangular grid out of two given one-dimensional arrays representing the cartesian indexing or matrix indexing

*dx,dy = fitzhugh_nagumo([X,Y],0,**param),* calling the function in t= 0
when plotting the vector field we can see the direction of the flow t

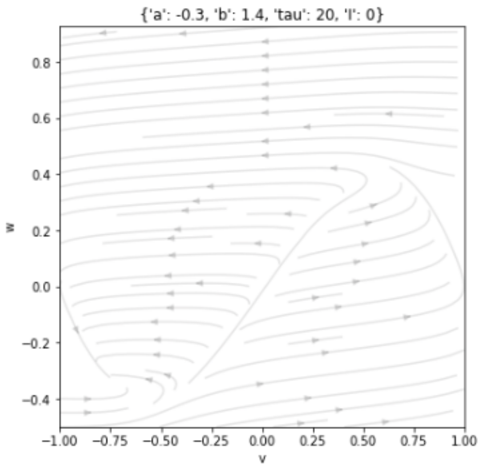


Figure 10

As we said before, whenever the external current is 0, the fixed point turns out a stable node. In figure we can observe how the direction of the flow follows a stable spiral
Around that fixed point between 0,5 and -1
Figure 10

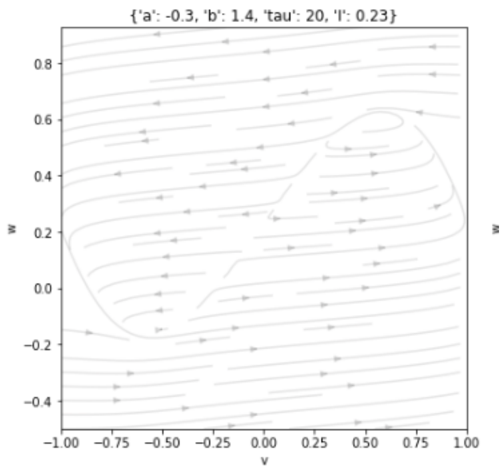


Figure 11

If the increased external current the action potential is produced and three fixed points will arise.
From left to right, stable focus, unstable focus and saddle node.
This is because the smallest perturbation will shift the system to the right and said points will vanish.
Figure 11

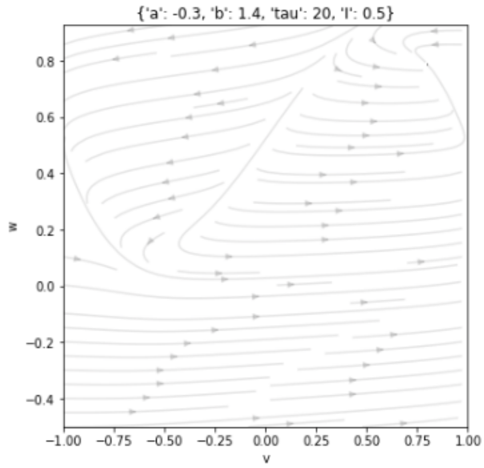


Figure 12

Again, the external current is enough to surpass the threshold thus the system is shifted to the right producing a hyperpolarization and the fixed point is a stable node , following the direction of the flow as a spiral , but now positive between 0,5-1
Figure 12

EQUILIBRIUM POINTS

The equilibria are found at the crossing between the null-isocline for v and the one for w.
Find the polynomial equation verified by the equilibria of the model.
First obtain a polynomial in v intersecting the null-clines.

$v - v^3 + I = \frac{v}{b} - \frac{a}{b}$;
$\left(\frac{1}{b} - 1\right) v + v^3 - \frac{a}{b} - I = 0$;
$f(v*) = 0 = v^3 + v * \left(\frac{1}{b} - 1\right) - \frac{a}{b}$;

From its coefficients obtain the roots.
with the function numpy.roots we find the roots of the polynomial, with np.real we only keep the real ones
the coefficients of the polynomial are:
1 for v^3
0 for v^2
$\frac{1}{b} - 1$ for v
And $-\frac{a}{b} - I$ for the independent term
For each v* we have a w value, if we remember from before $w = v - v^3 + I$, thus
$w *= v * - v *^3 + I$ for each v* then the position of the equilibrium;
$r - r^3 + I$

NATURE OF THE EQUILIBRIA

Obtain the Jacobian Matrix of the Flow:

$$J\Big|_{v,w} = \begin{bmatrix} \frac{\partial F_1(v,w)}{\partial v} & \frac{\partial F_1(v,w)}{\partial w} \\ \frac{\partial F_2(v,w)}{\partial v} & \frac{\partial F_2(v,w)}{\partial w} \end{bmatrix}$$

Taking into a account the null clines;

$v - v^3 + I - w$ =0

$v - a + bw = 0$

Then

$\frac{dF1}{dv}$= 1-3v^2

$\frac{dF1}{dw}$ = -1

$\frac{dF2}{dv}$ =1, in terms of τ $\frac{1}{\tau}$

$\frac{dF2}{dw}$=-b, in terms of τ $\frac{-b}{\tau}$

when implementing the list in Python:

np.array([[1-3v^2 -1],[$\frac{1}{\tau}$,$\frac{-b}{\tau}$]]), equations in Python code:

np.array([[- 3 * v**2 + 1 , -1],[1/tau, -b/tau]])

Once the previous code run, show in a Table the value the Jacobian for all the roots in all the scenarios. Compute also, its determinant and trace.

We can either calculate the stability of the equilibrium points by its eigenvalues or by the trace and determinant

By eigenvalues:

```
def stability(jacobian):
eigv = np.linalg.eigvals(jacobian),
```

calculating the eigenvalues

then we have to divide them by cases:

```
if all(np.real(eigv)==0) and all(np.imag(eigv)!=0;
        nature = "Center"
```

if one of them is real and equal to 0 and the other one is imaginary and not equal to 0 then the na ture of the equillibrium point is a center

```
elif np.real(eigv)[0]*np.real(eigv)[1]<0:
        nature = "Saddle"
```

if the multiplication of both real eigenvalues is negative then the nature of the equllibrium point is a saddle

```
else:
```

```
        stability = 'Unstable' if all(np.real(eigv)>0) else 'Sta
ble'
        nature = stability + (' focus' if all(np.imag(eigv)!=0) e
lse ' node')
```

if the real both eigenvalues are positive is stable
the rest of the cases the nature is stable
we define the nature of the equillibrium point as the stability
if all the eigenvalues are complex and different from 0 it is focus
else it is a node
in conclusión: stability + node or focus

on the other hand we can make the same división but using the trace and determinant:

```
def stability_alt(jacobian):
determinant = np.linalg.det(jacobian)
calculating the determinant of the jacobian matrix with the fucn
tion det
trace = np.matrix.trace(jacobian)
```

calulating the trace of the jacobian matrix with the funcgtion trace

dividing by cases:

```
if np.isclose(trace, 0):
        nature = "Center (Hopf)"

if the trace is 0 and the determinant postive we have a center

elif np.isclose(determinant, 0):
        nature = "Transcritical (Saddle-Node)"
```

if the determinant is negative and the trace is 0 we have a transcritical saddle node

```
elif determinant < 0:
        nature = "Saddle"
```

if the determinant is negative we have a saddle node

```
else:
        nature = "Stable" if trace < 0 else "Unstable"
        nature += " focus" if (trace**2 - 4 * determinant) < 0 e
lse " node"
```

for all the cases where the determinant is 0:
if the trace is negative it is stable, if positive unstable

the nature is defined as the stability plus:
if we are above the curve then it is focus
if we are bellow then it is a node

```
{0: ['Stable node'],
 1: ['Stable focus', 'Unstable focus', 'Saddle'],
 2: ['Stable node']}
```

$$\begin{bmatrix} 1 - 3v^2 & -1 \\ \frac{1}{\tau} & -\frac{b}{\tau} \end{bmatrix}$$

COMPLETE PHASE DIAGRAM

At this point, you should be able to plot the phase diagram including null-clines, equilibrium points and some trajectories from the equilibrium points when suffer from small perturbations. Show the phase diagram for all the scenarios. Explain in detail the obtained results for the trajectories depicted.

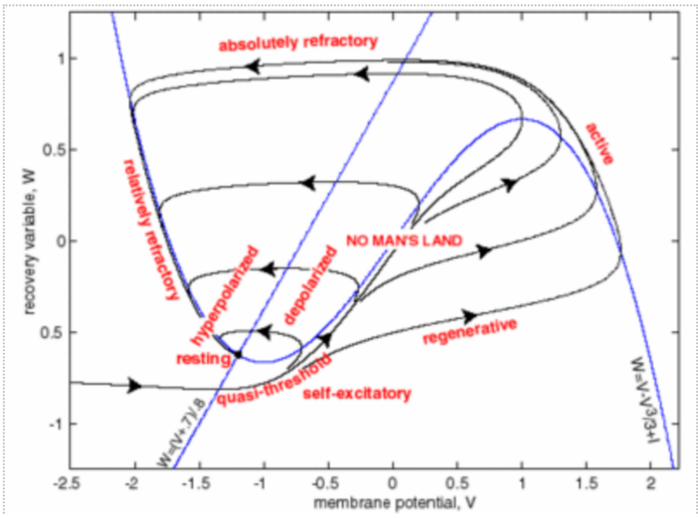


Figure 13

Phase portrait and physiological state diagram of FitzHugh-Nagumo model

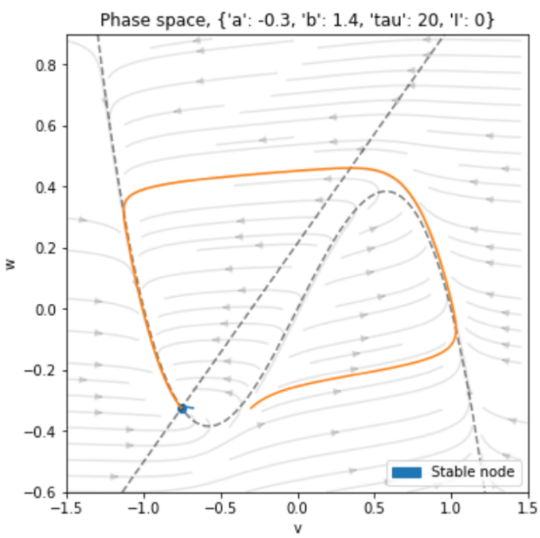


Figure 14

Absence of all-or-none spikes

When i is very weak or 0 the equilibrium is on the left branch thus the model is resting

As we can see the first graph (figure) corresponds to the stable node mentioned before, the flow goes to the right when the v nullclines are above the w nullclines and viceversa when its bellow.

Because the eigenvalues are negative and real the fixed point turns out a stable node

The trajectory of this fixed point is clearly a spiral sink or stable spiral around this said point

The FitzHugh-Nagumo model permits the entire solution to be viewed at once. This allows a geometrical explanation of important biological phenomena related to neuronal excitability and spike generating mechanism

Weak stimuli (small pulses of I > low I) result in small amplitude trajectories that correspond to the subthreshold responses

On the contrary strong stimuli result in large amplitude trajectories that correspond to suprathreshold

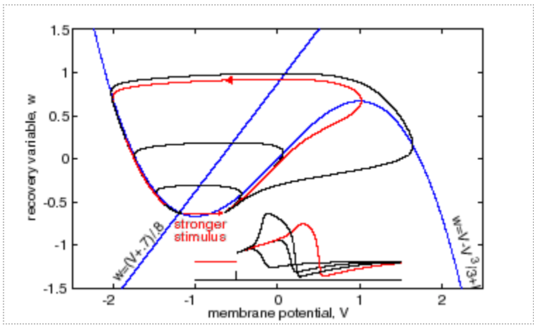


Figure 15

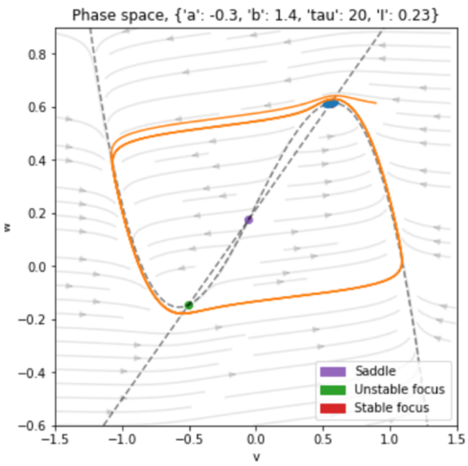


Figure 16

Excitation block
This model does not need to have a well-defined firing threshold
There exists a saddle point which make reference to a "saddle equilibrium", in other words a "quasi-threshold". This means that, the trajectory follows an unstable branch (Green point), so that all paths near this unstable point diverge rapidly away either to the right or left which produces a "all or none" mentioned before
In this case we have three equilibrium points that result from the intersection between the nullclines for v and the nullclines for w due to the action potential produced when the external current is enough to surpass the threshold.

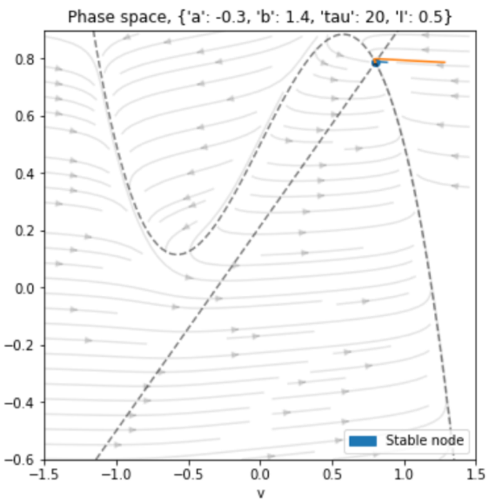


Figure 17

Over excitation block
The current increases, the nullcline will shift upwards and the equilibrium slides to the right on to the middle unstable branch
If we keep increasing the current will shift to the right, stable branch, the oscillations get immediately blocked, this is the reason why we do not see oscillations anymore (yellow line), because the oscillations disappear due to the over excitation
This phenomena is called hyperpolarization

BIFURCATION DIAGRAM

Now, we are going to plot the bifurcation diagram for v with respect to parameter I. Vary the paramenter I and b betweeen the specific extreme of the indicated interval.

ispan = np.linspace(0,0.5,200), creating an array between 0-0,5 with 200 points

bspan =np.linspace(0.6,2,200), creating an array between 0,6-2 with 200 points

BIFURCATION ON THE EXTERNAL STIMULUS I

Fill the following function to plot the bifurcation diagram for v with respect to parameter I.

*roots = find_roots(**param),* calling the function find roots with each parameter

*J = jacobian_fitznagumo(v,w, **param),* calling the function jacobian

nature = stability(J), calling the function stability

det.append(np.linalg.det(J)); appending the determinant of the jacobian matrix

trace.append(J[0,0]+J[1,1]); appending the trace of the jacobian matrix

Show the bifurcation diagram in your document. How many number of bifurcations can you detect?

As we can see we can detect 5 types of fixed points:

1) Stable node
2) Stable focus
3) Unstable node
4) Unstable focus
5) Saddle

What kind of bifurcation are they?

There are four bifurcations of codim 1 in this diagram:

two fold bifurcation (saddle-node) and two Hopf bifurcations (stable focus-unstable focus).

Show in your document the equilibrium trajectory in the Jacobian trace/determinant space. What kind of trajectory we find in this particular space?

Degenerate node

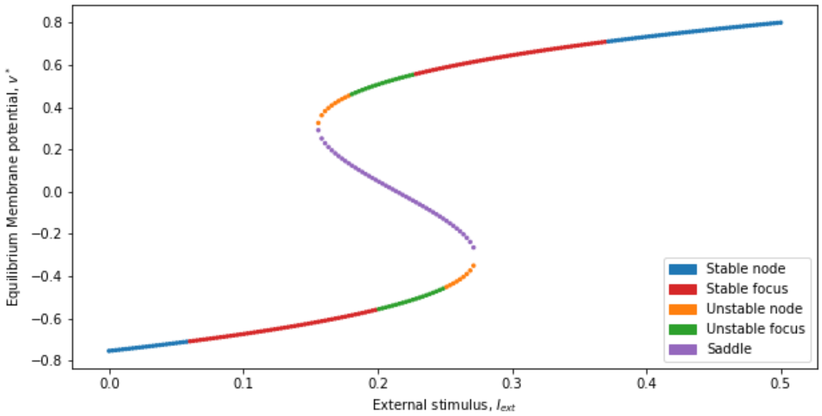


Figure 18

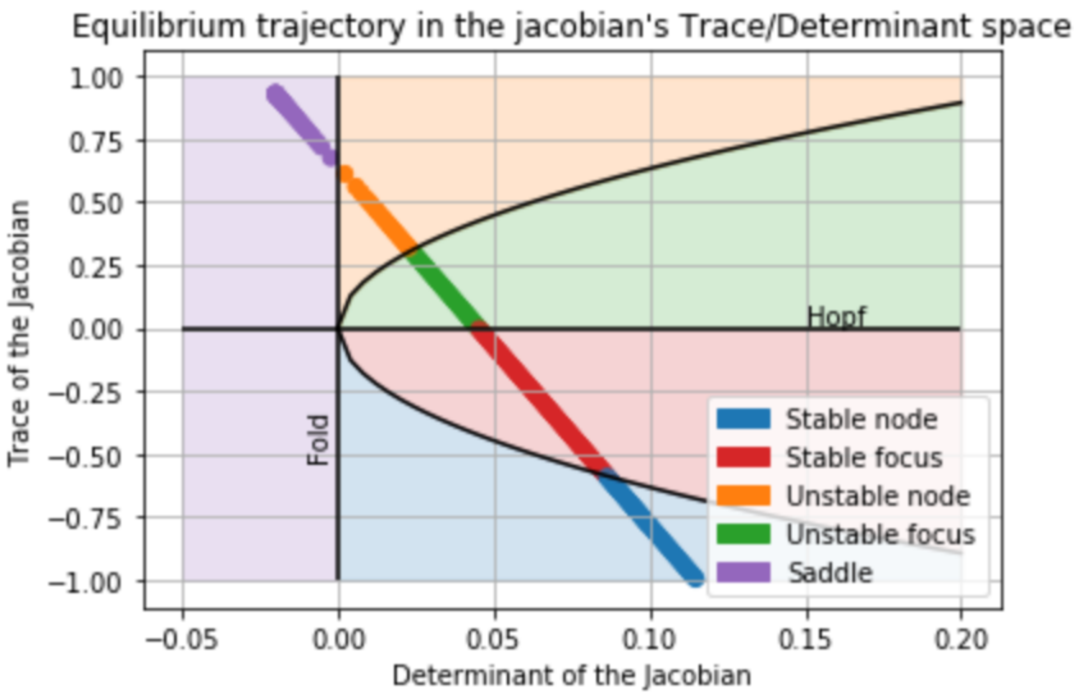


Figure 19

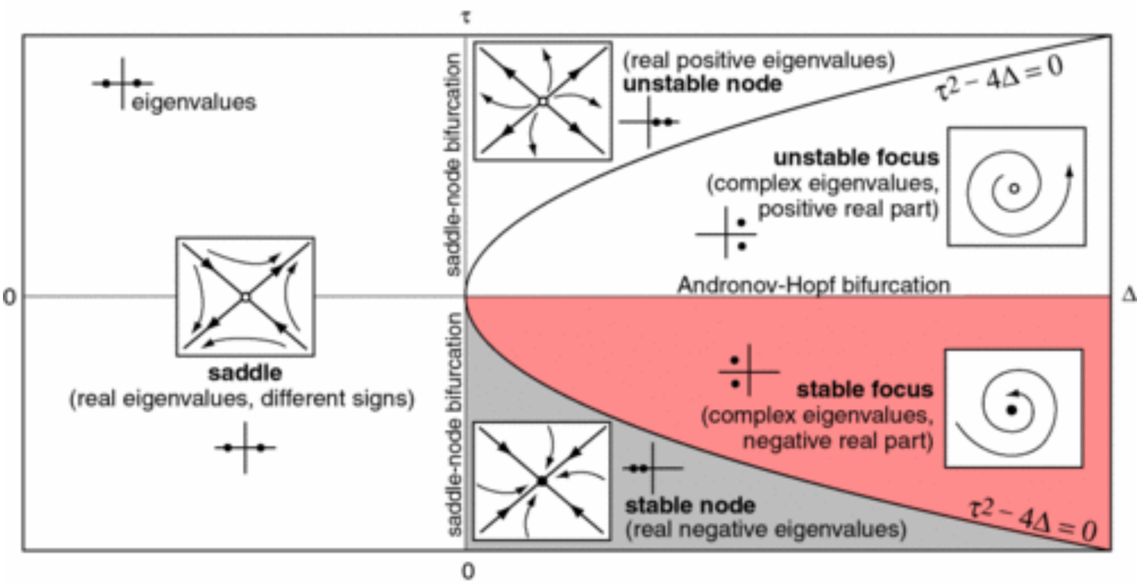


Figure 20

CODIM 2 BIFURCATIONS ON I AND B

Modify the following piece of code to depict the codim-2 bifurcation diagram for v with respect to the parameters I and b.

*r =find_roots(**param),* calling the function tu find the roots
*stab =[stability(jacobian_fitznagumo(v,w, **param)) for v,w in r],* stability of the jacobian matrix in the roots of the polynomia

Show the Codim 2 bifurcation on I and b. Analyze for what pair of values (I,b) the system will be periodic, non-periodic and mono-stable. Plot a trajectory in time and in the phase-space with the following conditions: I = 0,25 b = 1,2 a = −0,3 and τ = 20. Comment the obtained results.

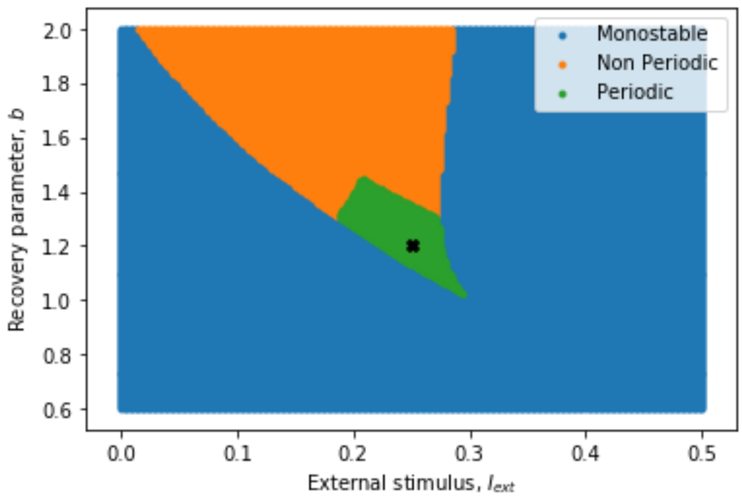


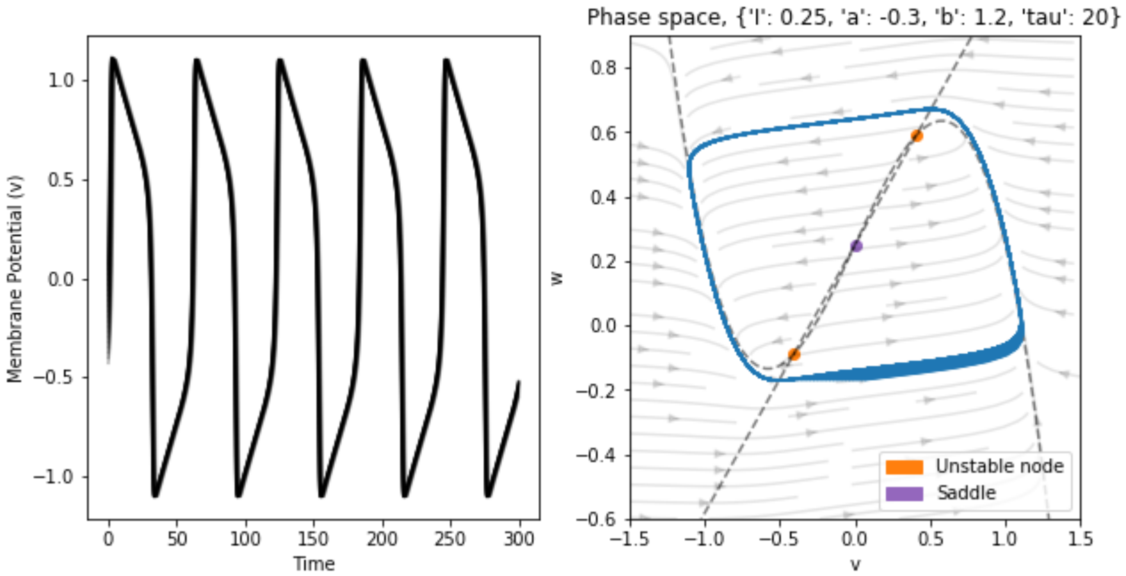
Figure 21

With a fixed b=1,4 we can see that for values of I [0,0.15] u [0.35,0.5] the system is monostable for example [0,1.4]
For values of I [0.15,0.2] we will have a non periodic system, for example [0.15,1,4]

For values of I [0.2,0.3] we will have a periodic system, for example [0.3,1.4]

The system is periodic when the length of the roots is not equal to 1 (I=0,23, b=1,4), and is not stable Non-periodic when the length of the roots if different to 1

If len ==1 mono stable (I=0, b=1,4)
Figure 21



Figures 23, 24

The graph comparing the time to the membrane voltage represents the steps of external current in said membrane. Starting at t=0, we are stimulating the system with sustained current step of amplitud I. Here the amplitude of the steps is less than the steps with a 0,5 external current what causes that even the smallest perturbations will diverge from the equilibrium. Figures 24, 24

NON-AUTONOMOUS SYSTEMS

Implement the non-autonomous version of Fitzhugh Nagumo model.

*X = np.array([x[0] - x[0]**3 - x[1] + I(t), (x[0] - a - b * x[1])/tau])* , the array must be of size 2.

The first argument is the derivative of V respect to time and the second argument the derivative of W respect to time beign v=x[0] and w=x[1], excitation and recovery variable respectively

Program step function 0 if t less than time and value is not:

return 0 if t<time else value

0 if t is less than time and the corresponding value when t reach consecutives time intervals
*return 0 if t<time else values[int(t//time)] if t<len(values)*time else values[-1]*
return a sine wave
*return magnitude * np.sin(freq * t)*

Show the eight trajectories programmed. There are 4 different stimulus initiated with 2 different initial conditions. Comment the results of each figure.

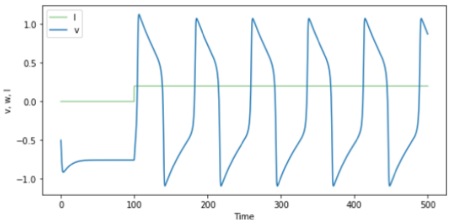


Figure 25

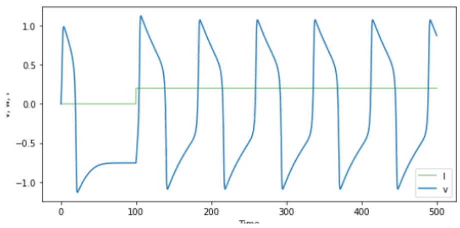


Figure 26

Step stimulus with a value of I= 0,2:
As we can see both systems are stimulated with a current in the form of a step, which causes a constant action potential.
The only difference between the two of them is the divergence on the initial conditions, the first on starting at -0.5 and the second one starting at 0

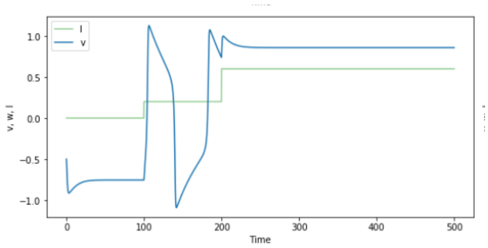


Figure 27

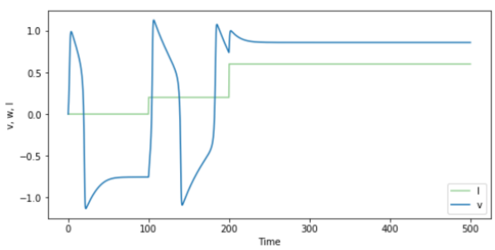


Figure 28

Step stimulus with values I= 0.1, I=0.2, I=0.6
This type of stimulation is escalated, it is easy to see how the time span increases 100 ms each time we stimulate with a new current. First we stimulate with a value of I=0.1 and
t= 0ms , then we stimulate with a value of
I= 0.2 and t=100ms , and a third stimulus of value I= 0.6 and t=200ms.
Again the only difference between both is the divergence on the initial condition, as the first one starts at -0.5 and the second one at 0.

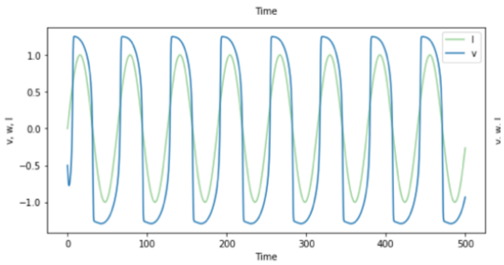


Figure 29

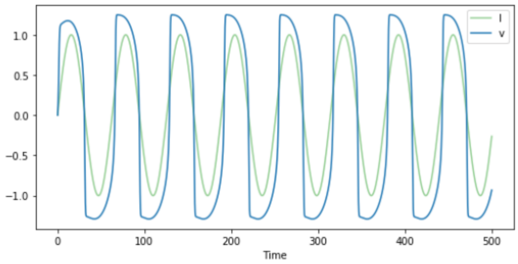


Figure 30

Periodic stimulus with a magnitude of 1 and a frequency of 0,1
This new type of stimulation is achieve with periodic stimulus such as a sin with amplitude of 1 and a frequency of 0,1:
Sin(0.1t).
The function sin oscillates between 1 and -1, thus with a frequency of 0.1 Hrz the sin will take values of 1 and -1 in an alternating way starting at t=0 ms.
The first one with an initial condition of -0.5 and the second one with an initial condition of 0

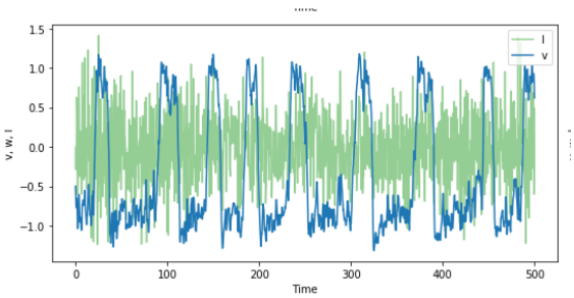


Figure 31

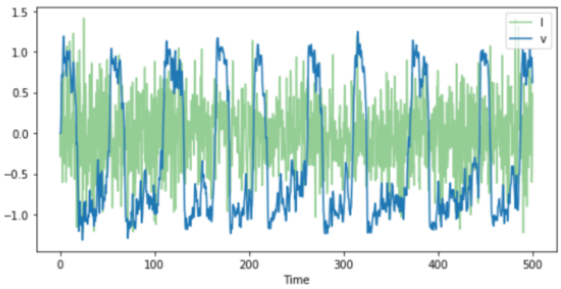


Figure 32

Noise in the form of spikes
This time the stimulation comes in the form of a noise which takes integer values starting at t=0ms and with two different initial conditions: 0 and -0.5

SOCHASTIC DIFFERENTIAL EQUATION

Complete the code to call the Euler-Maruyama ecuation:

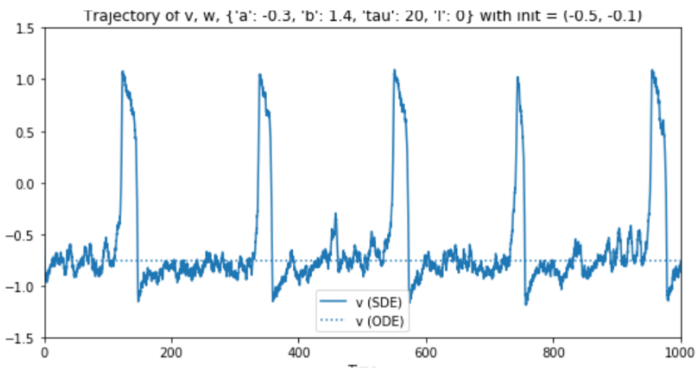
y[n] = (y[n-1] + flow(y[n-1],dt) * dt + noise_flow(y[n-1],dt) *np.random.normal(0,np.sqrt(dt));

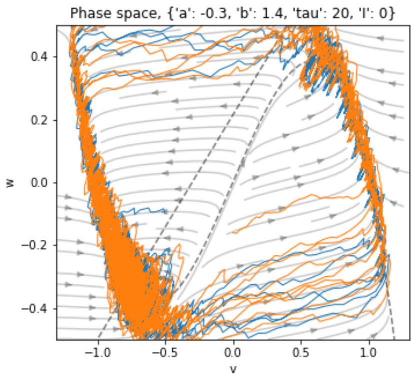
flow = partial(fitzhugh_nagumo, **param)
stochastic[i, j] = euler_maruyama(flow,noise_flow,y0=ic,t=time_s);

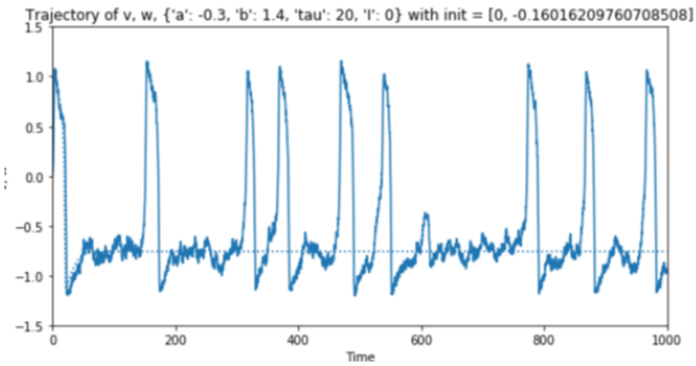
Show the final phase-diagram with the stochastic trajectories for the first two scenarios. Compare these diagrams with the obtained in the non-stochastic ODE.

A stochastic or random process can be defined as a collection of random variables that is indexed by some mathematical set, meaning that each random variable of the stochastic process is uniquely associated with an element in the set.

This time we have to consider the partial application of fitzhugh naguno, where Yt is the said random variable and B is the brownian motion (motion that particles follow in the fibers)

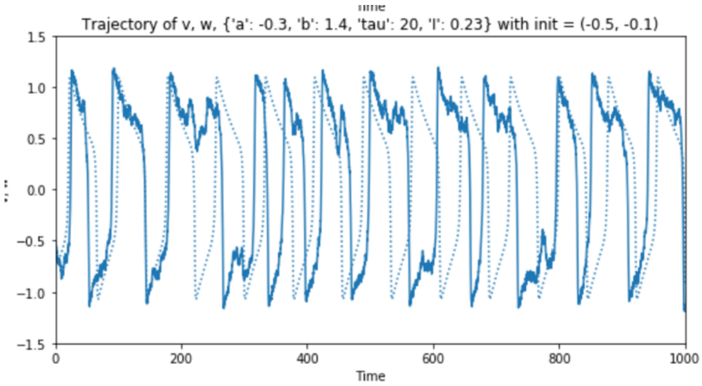


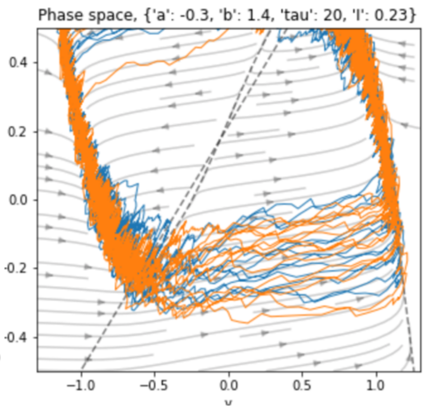


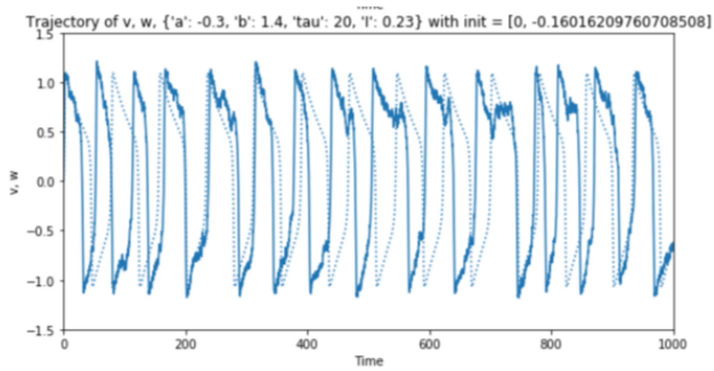


Figures 33, 34, 35

The difference is clear for I=0,
In the before case we did not have enough current to surpass the threshold so the phase point rapidly returns to p.
This time the application is partial so other parameters such as the noise flow which is an stochastic component will change drastically the graph as we can see in figures 33, 34, 35







Before we could see how when with a current enough to surpass the threshold so we obtained a series of steps each time we stimulated at t=0 ms
Now, as we are using the partial application, we need to take into account other component like the before mentioned, noise flow which is stochastic and provokes the perturbations we can see in dark blue, figures 36, 37, 38

Figure 36, 37, 38

References:

1. FitzHugh R. Impulses and Physiological States in Theoretical Models of Nerve Membrane. *Biophysical Journal*. 1961;1(6):445-466.

2. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*. 1952;117(4):500-544.

3. Amlan Barua (October 7, 2016), Understand the Dynamics of the FitzHugh-Nagumo Model with an App

4. https://neuro.bstu.by/ai/To-dom/My_research/Papers-2.1-done/Cognitive-S/Izhikevich-etc/FitzHugh-Nagumo%20mod.pdf