Università degli Studi di Padova

Department of Physics and Astronomy

Master Thesis in Physics of Data

Exploring the Interplay between Learning and Gene Expression

Master Candidate:

Sofía Pacheco García

Student ID **2070771**

Supervisor:

Anton Crombach
Inria de Lyon

Internal Tutor:

Samir Suweis
University of Padua

Academic Year
2023/2024

Abstract

In this study we present a mathematical model exploring the interaction between neuronal activity and gene regulatory networks in habituation learning. By integrating a leaky integrate-and-fire neuron model with a two-gene GRN described by continuous ordinary differential equations, we simulate how repeated stimuli lead to a diminished neuronal response, a key feature of habituation. Our results identify key gene interactions that are very convenient for this process given our model: an inhibitory gene suppressing the other gene, the latter being suppressed as well by stimuli and starting with an initial concentration equal to or greater than that of the inhibitory gene. Although the model effectively captures habituation, it simplifies aspects like synaptic interactions and stochasticity for tractability. Future work will refine these elements and validate the model with empirical data. Despite its limitations, this model provides a foundational approach to understanding the molecular mechanisms of learning and offers a basis for further research in computational neuroscience.

Acknowledgements

I would like to express my profound gratitude to my master thesis advisor, Anton, whose guidance and support have been instrumental in my journey into the research world. His mentorship, especially during moments when I felt stuck or lost, has been crucial, and I am eager to continue learning from him during my PhD. I am also deeply grateful to my research team, Beagle at Inria de Lyon, for their warm welcome and for fostering a collaborative and enriching environment. Special thanks go to Romain, a PhD student in the team, for his help with the parallelization of my code and for many insightful conversations during our breaks. I also appreciate the contributions of Clément Galan, whose internship code was a time-saver in generating network topologies. Additionally, I wish to thank my UNIPD supervisor, Samir, for introducing me to the field I have chosen for my future research and for his inspiring teaching. I am also thankful to the Master in Physics of Data at the University of Padova for providing me with a solid foundation in complex systems and data science, which is being essential for my academic development. Your support and collaboration have significantly enriched my first research experience, and I really appreciate it.

Contents

1 Introduction

Understanding the mechanisms behind learning and memory is a central challenge in neuroscience. These processes are critical since they enable organisms to adapt to their environments, which is essential for survival and for the development of complex behaviors. Despite the extensive research that has been going on in the field, the core mechanisms of learning continue to be an active area of study. Learning is typically associated with changes in synaptic strength, a phenomenon known as synaptic plasticity, which is influenced by neuronal activity. Empirical evidence suggests that neuronal activity is not only affected by but also directly influences gene expression, creating a feedback loop that is vital for adaptation and learning processes [1–4]. Synapse-to-nucleus communication mainly happens due to Calcium (Ca^{2+}) signals that are induced by synaptic activity and propagate into the nucleus [5]. Aditionally, there are some epigenetic markers that can regulate gene transcription dynamically in response to neuronal activation, allowing for long-term memory to be formed. Two examples of those epigenetic mechanisms are DNA methylation and post-translational modifications of histone tails [6]. Both the Ca^{2+} signaling and the epigenetic mechanisms are the foundational means for regulating gene expression in response to neuronal activity. This interplay between neuronal activity and gene regulation is increasingly recognized as a key factor driving long-term changes in neuronal function and behavior.

Habituation learning, recognized as one of the simplest and most fundamental forms of learning, provides a convenient framework for studying the underlying principles of learning processes. This non-associative learning phenomenon is characterized by a gradual decrease in behavioral responses to a repeated, non-threatening stimulus [7, 8]. Its simplicity makes it an ideal model for analysing the neural and molecular mechanisms involved in learning. For instance, habituation has been extensively studied in various organisms, including invertebrates such as the sea slug Aplysia. In Aplysia, habituation is evident through a decrease in the reflexive withdrawal response to repeated touches, which has been elementary in uncovering the role of synaptic plasticity [9]. Similarly, in insects like Drosophila melanogaster, habituation is observed as a reduction in response to repeated sensory stimuli, providing insights into sensory processing and memory formation [10]. In mammals, studies on rodents have demonstrated habituation as a decrease in exploratory behavior in response to repeated exposure to a new environment, highlighting the involvement of certain genes and nervous system structures [11]. These studies collectively emphasize the utility of habituation as a model for understanding the basic principles of learning and memory. In our work we will focus on this specific learning process.

Gene regulatory networks (GRNs) are complex systems within cells that govern the expression levels of genes, whose products have an important role in every process of life, for instance, in cell differentiation, in metabolism or in the matter we are delving into in this work: learning. While GRNs consist of genes, transcription factors, signaling molecules, and other regulatory elements, for simplicity, this report will focus only on genes when discussing GRNs. In neurons, GRN activity is closely linked to neuronal activity: neuronal stimuli induce specific gene expression patterns [12], which in turn modify neuronal function and influence behavioral responses. This dynamic interplay is crucial for understanding the mechanisms of learning at a deeper level. However, despite empirical evidence linking neuronal activity and GRNs, the exact mechanisms by which neuronal activity regulates specific GRNs, and how these GRNs influence learning, remain incompletely understood.

Consequently, mathematical modeling of a neuron-activity-gene coupling seems like a powerful approach to explore the contributions of gene regulation to learning and vice versa. This direction has been taken by few researchers [13] and is the one followed in this project. Through this kind

of approach, the aim is to narrow the gap between empirical observations and theoretical understanding, clarifying the processes underlying learning at a more mechanistic level.

In this study we develop a mathematical model that captures the bi-directional coupling between neuronal activity and GRNs, with a specific focus on habituation learning. We start by choosing an existing model for each of the two separate systems. On the neuron side, we work with the leaky integrate and fire model (LIF), which is a widely used, simplified mathematical framework for simulating neuronal electrical behavior [14]. It captures the basic features of how neurons integrate incoming signals and, based on that, produce spikes. A spike refers to an action potential, which is a fast and drastic change in the membrane potential due to it surpassing a threshold value because of integrated input signals. These spikes are the mechanism by which neurons transmit electrical signals along their axons to communicate with other neurons. During spiking, a neuron undergoes a series of phases: depolarization, hyperpolarization, and the refractory period, all driven by the movement of ions across the membrane. While the LIF model simplifies neuronal dynamics by focusing on a single variable, the membrane potential, it provides considerable advantages, including computational efficiency and ease of analysis. For the moment, these strengths make it preferable for us over more complex models, such as the Hodgkin-Huxley model, which, in spite of offering greater biological accuracy by incorporating ion channel dynamics, can be computationally intensive and analytically challenging.

For modeling GRNs, we opt for a continuous differential equation approach [15]. This model employs a system of Ordinary Differential Equations (ODEs), with each equation representing the change in concentration of a specific gene as a function of the concentrations of all genes within the network. Unlike logical models, which have a qualitative nature, continuous models provide insights into behaviors that depend on precise timing and specific molecular concentrations. This characteristic makes the continuous model particularly suited to the problem we aim to address. To simplify the model, we will not account for noise, which is why we also exclude stochastic single-molecule level models.

The adjacency matrix of the GRN constitutes an ensemble of parameters whose values are important for the model. We will sample a large amount of sets of parameter values that are representative for our parameter space and use this to build an atlas of GRN behaviors [16]. This atlas is a detailed compilation of data and visual representations that map out how different genes interact within the network, particularly in response to various stimuli or conditions. By analyzing this atlas, we can identify the characteristics our GRN must exhibit to accurately model habituation.

Finally, by integrating the LIF model for neuronal activity with the ODE approach for GRNs, we model the coupling to simulate both the immediate effects of neuronal activity on gene expression and the subsequent impact of these genetic changes on future neuronal responses. These changes in neuronal output are what we refer to as learning in this project. This approach contrasts with the conventional description and modeling of learning in the brain, which typically focuses on changes in synaptic strength between neurons [17].

The concept of learning as synaptic modification was pioneered by Donald Hebb [18], who introduced the well-known principle now known as Hebbian learning. This principle, often encapsulated by the phrase "cells that fire together, wire together", suggests that when one neuron repeatedly activates another, the synaptic connection between them strengthens. This increased synaptic weight, representing the efficiency of the synapse, increases the likelihood that the presynaptic neuron will successfully activate the postsynaptic neuron in the future. Synaptic modifications

constitute the basis for the storage of information and the adaptation of neuronal responses over time, with learning emerging from the collective interactions across neural networks.

In contrast, our study focuses on the dynamics of a single neuron as a simplified model for exploring the fundamental processes contributing to learning. We interpret external input signals as stimuli, which, in a more extended model, would correspond to inputs received via synaptic connections from other neurons. The neuron's output, characterized by action potentials, is modeled as being influenced by changes in gene expression within the neuron. We associate the changes in the output of the neuron, resulting from its coupling with gene expression, with the process of learning. This approach allows us to isolate and examine the interaction between neuronal activity and GRNs without the added complexity of synaptic interactions at this initial stage. While this simplification deviates from models that emphasize on synaptic plasticity, it offers a perspective on how gene expression might influence the output of individual neurons in response to stimuli. The long-term goal of this work is to expand the model to include synaptic interactions, integrating the contributions of both synaptic plasticity and gene regulation into a more biologically realistic framework for understanding learning.

Mathematical models, such as the one presented here, are highly valuable in the neuroscience community for two key reasons. Firstly, as emphasized, these models provide critical insights into how short-term neuronal signals translate into long-lasting cognitive functions, therefore enhancing our understanding of the mechanisms underlying learning. Secondly, they offer potential for developing targeted treatment strategies for cognitive disorders where these processes may be affected. For example, a pathology might arise from improper gene regulation in response to neuronal activity. A sufficiently accurate model, calibrated against empirical data, could help identify specific issues and enable the formulation of targeted treatment strategies. Moreover, shedding light on the molecular mechanisms of learning has broader implications beyond basic and applied neuroscience. It also informs for the development of artificial intelligence systems. By understanding how biological systems achieve learning, we can design more efficient and adaptable computational models.

In this study, we present a novel model of a neuron integrated with a two-gene GRN designed to simulate neuronal habituation behavior. Our mathematical model is complemented with GRN required characteristics derived from the constructed atlas of behaviors. In Section 2, the Methods, we detail the methodologies employed: the foundational mathematical models supporting our coupled model, the structure of the coupled model itself, the computational strategies used to achieve our simulation outcomes, and a thorough description of the GRN atlas construction. The Results section provides an analysis of the simulation results, comparing GRNs that support habituation with those that do not. Additionally, we include a second subsection analyzing the GRN atlas to identify the characteristics that enhance the likelihood of habituation learning within our model. Finally, Section 4 discusses the simplifications made in our approach and outlines potential future directions for this research.

2 Methods

2.1 Mathematical Models

As outlined in the Introduction, our goal is to develop a model that mathematically encapsulates the principle of habituation—an example of a learning process—by examining the impact of gene regulation on neuronal activity and vice versa. We will study a good way to integrate a neuronal model with a GRN model so that we can achieve our purpose.

In this section, we will explain the mathematical models for neurons and GRNs in which we will base our coupled model, followed by a description of the coupled model itself.

To clarify our approach, let us explain the kind of stimuli that we will pass to the neuron as external input. The external current to the neuron will have the form of a periodic step function $I_e(t)$ that simulates alternating periods of non-stimulation and sudden stimulation. The aim of the project is to build a model in which the neuron is able to habituate to those stimuli.

The idea is that the basic neuronal model will exhibit a strong response to the stimuli: the neuron will spike repeatedly whenever the external current is applied (we will call this spike rate behavior *stimulus spike rate*) and will cease spiking when the current is turned off (*basal spike rate*). Our goal is to develop a coupled model where the neuron's response to the alternating current demonstrates the phenomenon of habituation: initially, the neuron should react robustly to the first stimulus (when the current is first applied), but its response should diminish with subsequent stimuli. Although real stimuli can vary and animals may habituate to specific instances rather than all stimuli together, we will simplify the model by treating all stimuli in a general set where they are indistinguishable for clarity.

It is important to consider that the behavior of the system is likely to change depending on the selected GRN. Therefore, after developing the model, we will build a GRN atlas, testing the model on all combinations of parameters to identify the common characteristics that a GRN must possess for our model to simulate habituation learning.

2.1.1 Neuronal Model: Leaky Integrate and Fire Model

Among the various models that describe neuronal behavior, we have chosen one of the simplest: the Leaky Integrate-and-Fire (LIF) model [19, 20].

Integrate-and-Fire (IF) models are single-compartment models that describe a neuron's membrane potential using a single variable, V. The primary aim of these models is to capture how the flow of charges into and out of the neuron influences its membrane potential. Based on this concept, the simplest equation governing the membrane potential is given by that of an electrical circuit:

$$C_m \frac{dV}{dt} = \frac{dQ}{dt}$$

In this equation, V represents the membrane potential, which is the voltage difference between the inside and outside of the neuron. C_m denotes the capacitance of the membrane, analogous to a capacitor's ability to store charge in an electrical circuit. The variable Q is the charge stored on the membrane. One can notice that $\frac{dQ}{dt}$ corresponds to the current $I(t)$, or the rate of charge flow through the neuron's membrane. Therefore, this differential equation describes how the membrane

potential $V(t)$ evolves over time due to the current $I(t)$ flowing through the membrane capacitance C_m.

The current entering the neuron arises from both membrane and synaptic conductances, together in the single parameter i_m, in addition to any externally injected current I_e:

$$i(t) = -i_m(t) + \frac{I_e(t)}{A}.$$

Here, the transition from capital I (total current) to lowercase i (current per unit area) is achieved by normalizing the total current by the membrane area A. The membrane current i_m is given by the sum of contributions from each kind of ion channel j:

$$i_m = \sum_j g_j(V - E_j).$$

In this equation, g_j represents the conductance per unit area for channel type j, and $(V - E_j)$ denotes the driving force, with E_j being the reversal potential for the ion carried by channel type j. The current through these channels will be zero when $V = E_j$.

The reversal potential for a specific ion j can be computed through the Nernst equation:

$$E_{\mathrm{j}} = \frac{RT}{zF} \ln \left(\frac{[\ \mathrm{j}\ ]_{\mathrm{outside}}}{[\ \mathrm{j}\ ]_{\mathrm{inside}}} \right)$$

where R is the universal gas constant, T is the absolute temperature in Kelvin, z is the valence of ion j, F is Faraday's constant and $[j]_{outside}$ and $[j]_{inside}$ represent the concentrations of the ion outside and inside the cell, respectively.

If we don't explicitly add the biophysical mechanisms responsible for action potentials and instead simplify the model by grouping all membrane conductances into a single "leakage" term $i_m = \bar{g}_L(V - E_L)$, we can preserve the idea of modeling the neuron as an electrical circuit with resistor and capacitor in parallel. In this model, we define the externally applied current I_e as positive when it enters the neuron, and the membrane current i_m as positive when it exits the neuron.

We now multiply both sides of our equation by $r_m = 1/\bar{g}_L$, which represents specific membrane resistance. We define the membrane time constant as $\tau_m = c_m r_m$, c_m being the conductance per unit area. Additionally, we introduce the effective membrane resistance $R = r_m/A$ allowing us express r_m/A as a single parameter. Combining these definitions, we obtain the following equation:

$$\tau_m \frac{dV}{dt} = E_L - V + RI_e$$

The previous equation is only able to model subthreshold membrane potential dynamics. The LIF model is based on the behavior of neurons, which fire an action potential when their membrane potentials reach a threshold value V_{th}. After this action potential (spike), V is reset to $V_{rest} < V_{th}$. In order for our model to generate action potentials we will need to add a rule to our equation:

Whenever V reaches V_{th} an action potential is fired and V is set to V_{reset}.

Furthermore, real neurons go through spike-rate adaptation: the length of inter-spike intervals changes in time when a constant I_e is applied before arriving to a steady state value. This last

value models the fact that the probability for a neuron to fire is less right after a spike (which is called refractoriness). Our way to introduce this will be to augment the previous rule with the following forbidding condition:

Our neuron cannot fire for a certain period after a spike.

Taken all together and translated into mathematical language, our LIF model for a neuron can be writen as:

$$\begin{cases} \tau_m \frac{dV}{dt} = E_L - V + RI_e \\ \text{if } V(t) \geq V_{th} \implies \begin{cases} V(t) = V_{spike} & \text{depolarization} \\ V(t+\delta t) = V_{rest} - 10 \quad \text{with} \quad \delta t \to 0 & \text{hyperpolarization} \\ V(k) = V_{rest} \quad \text{with} \quad k \in (t, t + t_{refract}) & \text{refractoriness} \end{cases} \\ V_{rest} < V_{th} < V_{spike} \end{cases} \tag{1}$$

2.1.2 GRN Model: Continuous Differential Equation Model

To conceptualize our Gene Regulatory Networks (GRNs), we can express them as directed graphs, as suggested in [21], where nodes represent genes, and edges depict regulatory interactions. The adjacency matrix of these graphs is then employed in the mathematical model we will use to describe the genetic networks.

Among the possible ways to model an N-gene GRN [22], we choose to use Ordinary Differential Equations (ODEs), that allow to describe a more general and detailed model of regulation compared to other kinds of models [15]. More specifically, the model will be based on those introduced in [16, 23, 24] and it will consist of N differential equations, each describing the temporal evolution of the concentration of one specific gene depending on all the gene concentrations of the network.

$$\frac{dg^a}{dt} = R^a \Phi\left(u^a\right) - \lambda^a g^a$$

Here, on the left hand side of the equation, g^a stands for the concentration of gene a at a certain instant of time. Moreover, the two terms of the right hand side of the equation define regulated gene product synthesis and decay respectively. R^a represents the maximum synthesis rate of gene a, λ^a is the decay rate of the product of gene a, and $\Phi(u^a)$ is a sigmoid regulation-expression function that encapsulates the fundamental regulatory dynamics. Φ acts as a threshold function.

$$\Phi\left(u^a\right) = \frac{1}{2}\left(\frac{u^a}{\sqrt{(u^a)^2 + 1}} + 1\right)$$

where

$$u^a = \sum_{b \in G} T^{ba} g^b + E^a g^a$$

Let us clarify the parameters in this equation: T is the adjacency matrix of the graph, defining the interactions between genes, with its elements referred to as *regulatory weights* that take values in between ± 3.5. E, whose elements are also regulatory weights, represents the interactions between genes and external inputs. As it can be noticed, we have established that the effect of an external entity on a specific gene is self-regulated by the affected gene itself, thus, the external input will

have a greater effect on genes whose level of expression is greater. Regulatory weights can be positive (indicating activation of gene product synthesis), negative (indicating inhibition), or close to zero, which suggests no regulatory interaction.

In summary, the mathematical representation of our GRNs can be expressed as:

$$\begin{cases} \frac{dg^a}{dt} = R^a \Phi\left(u^a\right) - \lambda^a g^a \\ \Phi\left(u^a\right) = \frac{1}{2}\left(\frac{u^a}{\sqrt{(u^a)^2+1}} + 1\right) \\ u^a = \sum_{b \in G} T^{ba} g^b + E^a g^a \end{cases} \tag{2}$$

During this project we have chosen to keep things as simple as possible to focus on the modeling. That is why we are going to use two-gene GRNs.

2.1.3 Model Formulation for the Coupling

Having described the two systems that we intend to couple, the next step is to develop a model that integrates them, establishing a dependency between the two. Through this connection, we seek to create a mathematical framework that can effectively simulate the habituation process.

One-way coupling: the GRN is dependent on the neuron

Let us first focus on a one-way coupling: the idea is to establish a dependency of the GRN on neuronal activity. We aim for the effect of neuronal activity on genes to be significantly different (ideally, opposite) during periods of stimulation and non-stimulation. We seek this hoping that for some GRNs, the behavior of the neuron when a stimulus arrives induces a permanent change in the stable state of gene concentrations (g^a) with respect to the one they would reach if no stimulus had been given. A permanent change on the attractors of g^a could be linked to some kind of learning. To achieve this, we will focus on spike rates in the neuronal model, since those related to periods of non-stimulation will be minimal while spike rates related to periods of stimulation will be maximal. Regarding the GRN, the external input parameters E^a in its equation provides a straightforward place to make the link to neuronal input. Therefore, we will make GRN's external inputs (E^a) depend on the spike rate of the neuron at each time window: $E^a(r)$ with r the spike rate of the neuron in a certain time-window.

To ensure that basal and stimulus-induced spike rates have opposing effects, and to rescale these spike rates so that they fall within the same range as the other regulatory weights in the interaction matrix T, we transform the spike rates using a sigmoid function with extreme values of ± 3.5 before plugging them into E^a. This transformation allows us to assign negative values to spike rates near zero, which correspond to basal spiking behavior, and positive values to spike rates approaching the maximum possibly observed spike rate within the specified time window, reflecting stimulus-induced spiking behavior. The computation of the maximum spike rate is done by knowing the spike time scale, which takes into account depolarization, hyperpolarization and refractory period of the membrane potential: if we know that the total process of one spike takes a time τ_{spike}, then $r_{max} = \frac{l_{\text{window}}}{\tau_{spike}}$ with l_{window} the length in seconds of our chosen time window.

Then, mathematically, we have:

$$r_{rescaled} = \frac{3.5 \cdot x}{\sqrt{1+x^2}} \quad \text{with} \quad x = r - \frac{r_{max}}{2}, \quad r_{max} = \frac{l_{\text{window}}}{\tau_{\text{spike}}} \tag{3}$$

If we applied the result of the previous equation to the gene's external inputs without further modifications, it could be inferred that basal spiking inhibits genes, while stimulus spiking activates them. However, this interpretation is arbitrary—why not the other way around? Additionally, it is random to assume that all genes in a GRN are affected similarly by spike rate. To avoid this subjectivity, we will allow spike rate to have three different possible ways to impact on each gene:

$$E^a = \begin{cases} +r_{rescaled} \\ -r_{rescaled} \\ 0 \qquad\qquad \text{(non dependent)} \end{cases} \tag{4}$$

and we will try, given a specific N-gene GRN with its adjacency matrix, all possible combinations of E^a, $a \in [1, \ldots, N]$. Then by checking results for GRNs with different parameters, we will be able to see which parameter configurations help us get the behavior we are looking for.

From now on, we will call *parameter configuration* to the 6 number combination of a specific set of parameters for a GRN's adjacency matrix T and a specific pair of E^a.

Two-way coupling: the GRN is dependent on the neuron and vice versa

Focusing on the specific GRN configurations for which the previous one-way coupling yields a change in the attractors of g^a, we will now model the coupling back: how does a neuron depend on its GRN in our model?

Let us recall that our idea is to model a kind of neurons which get used to stimuli and react less and less to them. A possible way to insert this kind of behavior is by modifying V_{th}'s value depending on the gene concentrations. We can make V_{th} increase as our gene trajectories separate from the ones they would be following if the current applied to the neurons was a basal current all along time, and make it decrease when the two trajectories get closer to each other. In this way we expect that in the cases in which the attractor seems to have changed (even if stimuli still perturb stability), there is a permanent change in V_{th} as well, which will make the neuron spike only for greater values of membrane potential in the future. Therefore, a stimulus like the one we were applying might not be able to overcome V_{th} and won't make the neuron spike anymore.

Biologically, this could mean that inside our neurons we have two GRNs with same adjacency matrix, one of them with external inputs related to the actual spike rate of the neuron when the external current arrives just as we have modeled ($r_{rescaled}$ is computed with $r = r_{stimuli}$), and the other one with constant external input to both genes associated to a basal spike rate. In this second GRN, $r_{rescaled}$ is computed with

$r = r_{basal} = 0$, which mimicks the behavior that the genes of the first GRN would have if the stimuli never arrived.

Let us present the mathematical equation for V_{th} that models such coupling:

$$\frac{dV_{th}}{dt} = M \sum_{a \in G} \frac{d|g^a_{stimuli} - g^a_{base}|}{dt} \tag{5}$$

$$V_{th}[0] = V_{th\_const}$$

where $g^a_{base}(t)$ corresponds to the concentration of gene a at a certain time t if the input current was, and had always been $I_e = 0$, and $g^a_{stimuli}(t)$ is the concentration of gene a at time t in a system with alternating external current to the neuron. Moreover, M is a multiplicative factor that allows to tune the strength of the effect that we want gene concentrations to have on the threshold potential.

All together, the model can be summed up in the following set of equations:

$$\text{Model:} \begin{cases} \text{Neuron:} \begin{cases} \tau_m \frac{dV}{dt} = E_L - V + RI_e \\ \frac{dV_{th}}{dt} = \sum_{a \in G} \frac{d|g^a_{stimuli} - g^a_{base}|}{dt}, \quad V_{th}[0] = V_{th\_const} \\ \text{if } V(t) \geq V_{th} \implies \begin{cases} V(t) = V_{spike} & \text{depolarization} \\ V(t+\delta t) = V_{rest} - 10 \quad \text{with} \quad \delta t \to 0 & \text{hyperpolarization} \\ V(k) = V_{rest} \quad \text{with} \quad k \in (t, t + t_{refract}) & \text{refractoriness} \end{cases} \\ \text{with:} \\ V_{rest} < V_{th} < V_{spike} \end{cases} \\ \text{GRNs:} \begin{cases} \frac{dg^a}{dt} = R^a \Phi\left(u^a\right) - \lambda^a g^a \\ \text{with:} \\ \Phi\left(u^a\right) = \frac{1}{2}\left(\frac{u^a}{\sqrt{(u^a)^2+1}} + 1\right) \\ u^a = \sum_{b \in G} T^{ba} g^b + E^a g^a + h^a \\ E^a \in \{+r_{rescaled}, -r_{rescaled}, 0\} \\ r_{rescaled} = \frac{3.5 \cdot x}{\sqrt{1+x^2}} \quad \text{with} \quad x = r - \frac{r_{max}}{2}, \quad r_{max} = \frac{l_{\text{window}}}{\tau_{\text{spike}}} \end{cases} \end{cases} \tag{6}$$

2.2 Computational Implementation of the Model

In this section, we will explain the computational implementation, selection of parameters and time scales for the algorithms associated with the described models. This will help one understand how the figures in the result section are generated.

We have used Euler's first order differential method for solving all the ODEs.

2.2.1 Neuron Details

The values showed in **Table 1**, which are the ones used in [19] and are based on empirical evidence, are the ones chosen for the parameters of the neuron.

τ_m	0.01 s
E_L	-65 mV
V_{rest}	-65 mV
V_{th_const}	-50 mV
R	10 MΩ
$t_{refract}$	0.008 s

Table 1: Parameter values for neuronal model.

The membrane potential of the neuron is computed with a time-step of 0.001 s. This implies that within an interval of 10 time-steps (10 ms) we will have at most one spike, which agrees with the empirical fact that the spiking time-scale is of the order of 10 ms. For all our simulations, we compute membrane potentials for a time interval of 1000 s.

The current provided to the system takes values of 0 µA and 4 µA repeating five times the sequence: 90 s of $I_e = 0$ µA, 20 s of $I_e = 4$ µA and again 90 s of $I_e = 0$ µA. As mentioned before, it is built in that way so that we can link zero-current intervals to periods of time in which the neuron is not stimulated, and non-zero-current intervals to periods in which the neuron receives a high enough external input and, as a consequence, starts spiking. This is only an approximation, in reality it is difficult to find neurons completely at rest for a considerable amount of time even when they don't receive stimuli; basal spiking is this low-spike-rate behavior of neurons in absence of stimuli that we simulate as null-input current periods. Basal spiking is usually refered to in literature as *spontaneous activity* of the brain. Moreover, we choose the non-zero-current to have a value of 4 µA, because such a value makes the membrane potential overcome the theoretical (constant) potential threshold. Finally, together with the previous parameter values, we impose the initial condition $V_0 = V_{rest}$.

2.2.2 GRN Details

In the case of the GRNs, the values given to the parameters are collected in **Table 2**.

R^a	1
λ^a	0.05
T^{ab}	(-3.5, 3.5)

Table 2: Parameter values for the GRN; while the three first are completely fixed (arbitrarily), the interaction matrix parameters can adopt values from the given interval.

Gene concentrations are computed in time steps of 0.1 s. This allows us to make several time steps of the neuron have an effect on one time step of the genes, which is useful for the coupling: the neuron's spike rate is computed along a period of 0.1 s, which is directly input to the E^a parameters of the genes in a certain time step. Again, as in the neuron case, we have used a time interval of 1000 s.

The initial conditions are also given when generating gene concentration trajectories.

2.2.3 Coupling Details

As explained above (Section 2.1.3), the external inputs of the GRN (E^a) depend on the spike rate of the neuron computed in 0.1 s time windows. Within windows of such length, a neuron will fire 10 times at most, since the spiking time scale is of 0.01 s. Spike rates are computed in consecutive windows that do not overlap with each other. So we have values for two more parameters:

l_{window}	0.1 s
τ_{spike}	0.01 s

Table 3: Parameter values for the coupling.

It is also important to mention that the time scale of gene regulation is of the order of seconds, minutes or hours (we will take it to be of about 10 s), in contrast to that on neural response (spikes) which is of 0.01 s. To take care of this difference, we will consider that the influence of GRN on neurons has a delay of 10 s. Thus, given a spike rate at time t of the neuron, its effect is immediate on the GRN. In turn, gene concentrations' impact on the neuron's V_{th} will happen with a delay of 10 s.

2.3 Atlas of GRNs

The previous model can be applied to GRNs with different parameter configurations. Our goal is to generate results for a representative set of GRNs so that, after, we can separate the instances in two clusters (those that allow learning and those that do not) such that we can draw conclusions about specific characteristics that our neuronal genetic network should possess to enable the modeling of habituation. As mentioned in the Introduction, for simplicity, we choose to work with two-gene GRNs.

In this section we will explain how we generate an atlas of GRNs based on the results or behaviors yielded once they are plugged into the model's algorithm. We have six parameters to choose: four corresponding to the network's adjacency matrix, which will be drawn using a sampling method explained in Section 2.3.2; the two other parameters correspond to the external input relationship with the neuron. For a specific adjacency matrix of the GRN, we generate all the possible combinations of these last two parameters (E^a with $a = 0, 1$), knowing that each of them can take the values shown in Eq.(4).

Let us define the terms that we will be using:

- ⋆ **GRN topology:** qualitative information of a GRN's adjacency matrix that only considers whether a parameter is positive (activating interaction), negative (inhibiting interaction) or zero (no interaction).

⋆ **GRN parameter set:** final adjacency matrix of the GRN containing the information of the network's topology and the intensity of each parameter. Inhibitory parameters can take values among $[-3.5, 0)$ and activatory ones can take values among $(0, 3.5]$.

2.3.1 Network Topologies Generator

The first step concerning the GRN atlas is to generate all the possible GRN topologies for a two-gene GRN.

To begin with, we enumerate all the possible combinations of the numbers $-1, 0, +1$ in groups of four. Remember that -1 corresponds to inhibition, 0 to no impact and $+1$ to activation on the interaction matrix. The order matters to us and numbers can, of course, be repeated. Therefore, the problem reduces to the computation of the number of possible permutations with repetition and the total number of combinations is given by $3^4 = 81$.

Eliminating isomorphisms in adjacency matrices is essential when one wants to ensure that structurally identical (isomorphic) graphs are counted only once. That is, indeed, our next step. We do this by checking with the function `nx.is_isomorphic()` in the `NetworkX` library.

After this process, we end up with a set of 39 topologies for the two-gene network.

2.3.2 Adjacency Matrix Parameter Search: Latin Hypercube Sampling

The next step is to sample parameter sets for each of the network topologies that were generated following the guidelines on section 2.3.1. In spite of the simplest procedure being random sampling, we decided to use the Latin Hypercube Sampling method (LHS). In the following paragraphs we will explain why.

With random sampling we cannot be sure that elements from all subsets of the sampling space will be generated. This makes the technique susceptible to missing the sampling from important subsets of the sampling space which have a low probability but high consequences. Stratified or importance sampling provides a way to avoid this problem by specifying subsets of the sampling space (strata) from which samples will be selected. However, as the dimensions of the sampling space get higher, determining the strata and their probabilities becomes a significant challenge. LHS can be seen as a balanced approach that combines the advantages of both random sampling and stratified sampling [25]. We will use this last method in order to explore our parameter space efficiently.

To illustrate the general procedure of Latin Hypercube Sampling (LHS) [26], consider a scenario where N denotes the number of samples desired and d represents the number of dimensions. In LHS, each dimension is divided into N equal intervals, referred to as *strata*. For each dimension, one value is randomly selected from each interval, ensuring that every interval is sampled once. This leaves us with a set of values for each dimension. Following this, the order of the samples within each dimension is shuffled to achieve unbiased distribution across the intervals. Finally, the sample points are constructed by combining the sampled values from each dimension, with each combination forming a unique point in the multidimensional space.

In this study we take $N = 50$, so we have 50 samples for each network topology. Additionally, $d = 4$, corresponding to the four parameters in the interaction matrices.

2.3.3 GRN's External Input Relationship to the Neuron

Up to now we have 39 network topologies and 50 sampled parameter sets for each of them, that is 39×50 GRN adjacency matrices in total. Now, to each of those, we want to assign all combinations of external input relationships to neurons. If we combine in groups of two with repetition the possible values of E^a in equation (4), we have 9 combinations. The pair formed by $E^0 = 0$ and $E^1 = 0$ is not interesting to us, because of the definition of the coupling: this combination will never give rise to a change in the stable state concentration of the genes and thus, we will never be able to model habituation with it. Therefore, we have $9 - 1 = 8$ possible $E^0 - E^1$ combinations. That gives us $39 \times 50 \times 8$ GRNs for our atlas.

2.3.4 Initial Concentrations

There exist other two parameters that have not been mentioned yet: the initial concentrations of the genes. Given the parameter values chosen for our gene equations, their concentrations are bounded between the values [0, 20]. The initial concentrations that we impose must, therefore, be constrained by those values too. We decided to take five possible evenly spaced initial conditions (i.c.) for each of the genes. Given that a complete set of i.c. consists of a pair of concentrations, each belonging to one of the genes, with our choice we will have 25 possible combinations of i.c., which means 25 sets of initial concentrations.

2.3.5 Parallel Computing

All in all, the number of times that we will let our system evolve is $39 \times 50 \times 8 \times 25 = 390\,000$. The computational time required for doing so was very high, of the order of days. That is why we decided to use our computing resources (a remote machine with multiple cores) and parallelize our code. For this purpose we use `concurrent.futures.ProcessPoolExecutor`. This Python module is designed to execute functions asynchronously using a pool of separate processes, making it ideal for CPU-bound tasks (tasks that are limited by the speed of the CPU). It runs tasks in parallel across multiple CPU cores by using separate processes, which allows one to take advantage of multi-core processors. With its help we have reduced the computation time of our problem from days to the order of hours. Bellow we elaborate on the technical details.

We first divide our data in smaller and more manageable pieces or *chunks* of size $n_{\text{samples}} \times n_{\text{external\_input\_relationship\_to\_neurons}} \times n_{\text{i.c.}}$. In our case that is $50 \times 8 \times 25 = 10\,000$ instances, which leaves us with 39 chunks. Then, with the tools provided by Python's module and choosing a number of workers, each of the chunks is processed separately.

The output of this process is a table with as many rows as parameter configurations with specific initial conditions that we have. The columns specify all the parameters that characterize each instance (network topology, parameter set, external input relationship to the neuron and initial conditions), plus a last column that contains the features we want to use afterwards for the clustering. These features can be computed from all the data obtained while leaving the whole system progress over time, this includes gene concentrations along time, membrane potential along time, potential threshold, etc.

2.3.6 Hierarchical Clustering

Let us remember the objective of this study. Our aim is to classify instances into two distinct groups: those that can lead to modeling habituation and those that cannot. We propose using

hierarchical clustering as a viable method for this task. Hierarchical clustering builds clusters iteratively, where each new cluster is formed from previously established clusters, resulting in a dendrogram, a tree-like diagram that illustrates the hierarchical relationships among all clusters. This graphical representation enables us to visualize the nested structure of clusters and their interrelationships.

For effective clustering it is essential to select the most relevant features that align with our objectives. Providing these key features will guide the algorithm in forming the desired clusters. In our analysis, we determined that a single feature suffices for our purpose: the number of spikes within the time window. This count is expected to be high in the absence of habituation and decrease as the model increasingly captures the habituation effect. We use the module `scipy.cluster.hierarchy` of the SciPy library to perform our hierarchical clustering.

3 Results

In this section, we present and discuss the figures that summarize our findings. First, we detail the results obtained from simulating the systems described in the Mathematical Models section. These results are presented for two scenarios: one in which our model predicts habituation and one where habituation is not observed. In the second part, we introduce our GRN atlas, which enables us to categorize the data into two distinct groups—habituation and non-habituation. We then analyze the implications of these classifications.

3.1 Simulation Outcomes

The external current input to our neurons is the one showed in **Figure 1**. Such a setup simulates alternating intervals of non-stimulation and stimulation.

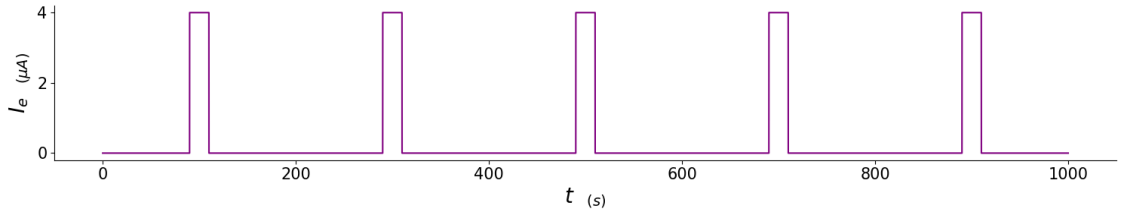


Figure 1: Input current to the neuron along time, it looks like a periodic step function taking values 0 µA and 4 µA.

Using the neuron's equations Eq.(1), one can check the effect of the external current when it is given to the uncoupled neuron (**Figure 2**). As we expected, the neuron spikes whenever the current is set to the high value and remains calmed when the input current is turned off, since in that case the membrane potential does not go over V_{th}.

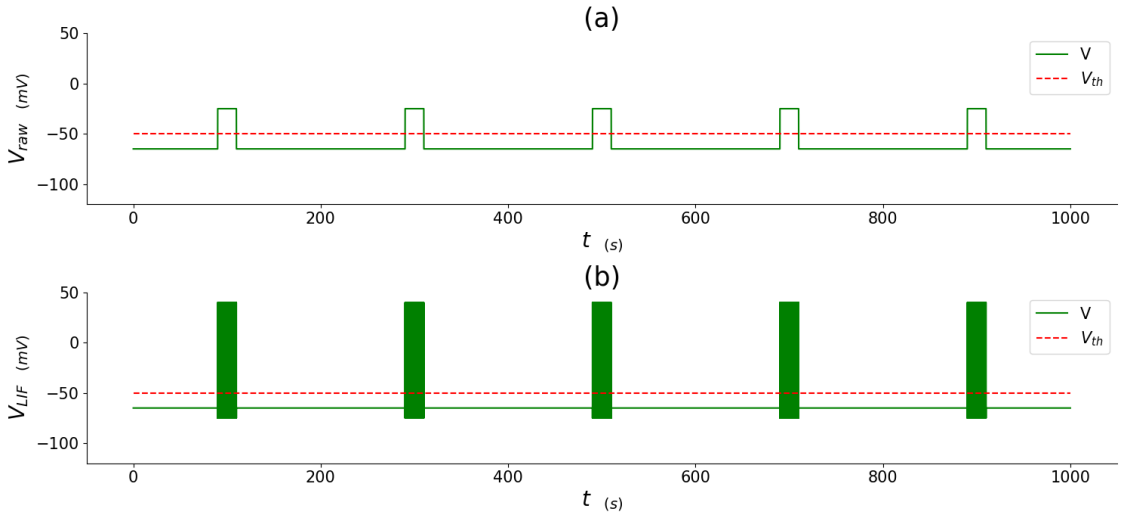


Figure 2: **(a)** Raw membrane potential of the neuron when the current in **Figure 1** is input; these values have been computed with the formula that only takes into account subthreshold potential dynamics in order to be able to see how the potential surpasses V_{th}. **(b)** Leaky Integrate and Fire model membrane potential for a current as the one in **Figure 1**.

In order to show the coupling simulation results and to compare them with the uncoupled

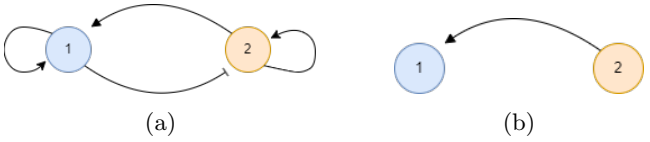
behavior of the systems, we choose two GRN configurations with specific values for their initial concentrations g^a_{ini}. The first GRN corresponds to an instance that presents habituation and the second one to a non-habituation case.

The parameters for the GRNs whose results we present in this section are:

$$T_{hab} = \begin{pmatrix} 2.4 & 2.5 \\ -2.9 & 2.9 \end{pmatrix}, \quad (E^1, E^2)_{hab} = (+r_{rescaled}, -r_{rescaled}), \quad (g^1_{ini}, g^2_{ini})_{hab} = (0, 0)$$

$$T_{non-hab} = \begin{pmatrix} 0 & 3.25 \\ 0 & 0 \end{pmatrix}, \quad (E^1, E^2)_{non-hab} = (+r_{rescaled}, +r_{rescaled}), \quad (g^1_{ini}, g^2_{ini})_{non-hab} = (0, 10)$$

where, in T, row i contains the information on how gene i is affected by the rest of the genes in the network while column i contains the information on how gene i affects the rest of the genes in the network. If we assign the arrow symbol $\longrightarrow$ to any activatory interaction and the T-bar symbol $\dashv$ to any inhibitory one, we yield two graphs like the ones in **Figure 3**:



(a) (b)

Figure 3: GRNs with the adjacency matrices we are working with. (a) corresponds to the case with habituation and (b) to the non-habituatory case.

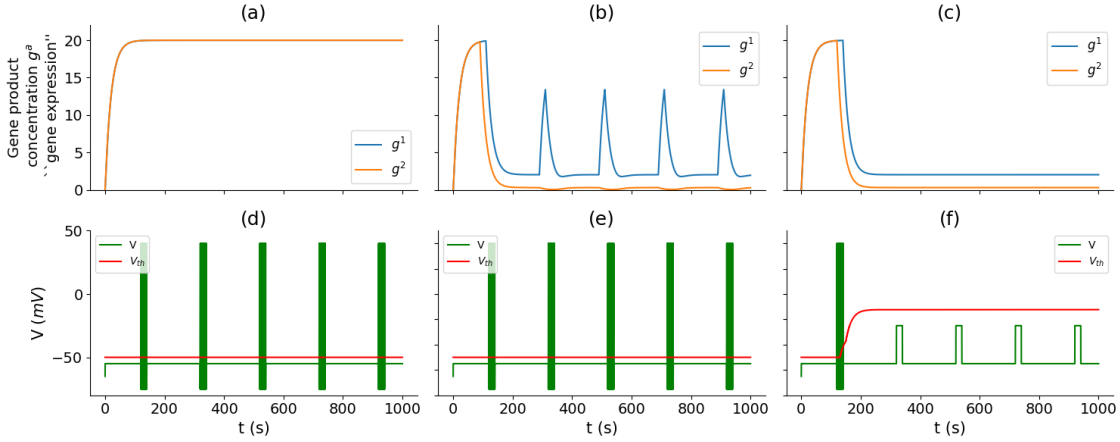


Figure 4: Simulation outcomes for the habituation case. (a) Concentration trajectories for the two genes when the neuron and the GRN are uncoupled and the neuron receives the external current in **Figure 1**. **(b)** Concentration trajectories when the GRN depends on neuronal spike rates and the neuron receives the external current in **Figure 1**. **(c)** Concentration trajectories when the GRN is dependent on the neuron ($E^a(r_{stimuli})$) and the neuron is dependent on the GRNs ($V_{th}(r_{base}, r_{stimuli})$) and the neuron receives the external current in **Figure 1**. **(d)** Membrane potential and potential threshold for the neuron corresponding to case (a), when there is no coupling with the GRN. **(e)** Membrane potential and potential threshold corresponding to case (b), for the one-way coupled system (the neuron does not depend on the GRN). **(f)** Membrane potential and potential threshold corresponding to case (c), for the two-way coupled system (V_{th} depends on g^a).

In **Figure 4** we can see the results for the habituatory case sorted from less to more coupled systems. In **(a, d)** we can see the GRN and neuron's behavior respectively when the two systems are uncoupled, following equations (1) and (2). The GRN trajectories in that figure are analogous to the concentrations we would get if the GRN depended on the neuron through (3) and (4) and this last one did not receive any input current and ($I_e = 0\,\mu A \implies$ no stimuli $\implies$ no spikes). In **(b, e)** we have the figures yielded after the one-way coupling (genes depend through their external inputs on neuronal spike rates but the neuron is independent on the GRN). We can see how the first stimulus already changes the attractor of the two genes, which goes from $(g^1_{ini}, g^2_{ini}) = (20, 20)$ in **(a)** to $(g^1_{ini}, g^2_{ini}) = (0, 3)$ in **(b)** approximately. However, there is still a strong reaction of gene concentrations to the next stimuli, which makes the stable state concentrations be perturbed, since the neuron keeps on spiking for it not being dependent on gene concentrations. In **(c, f)** both systems are dependent on each other. One can appreciate that, in this case, the gene concentrations are not perturbed anymore after the first stimulus. That happens because the neuron's potential threshold V_{th} is now dependent on how fast the gene concentrations separate from those in **(a)** (which are the g^a we would have if no external input had been applied). The dependence is made through (5). We can see how V_{th} increases, given the change of attractor on gene concentrations, not allowing the membrane potential to overcome the potential threshold and, therefore, not allowing the neuron to spike or react to future stimuli anymore **(f)**.

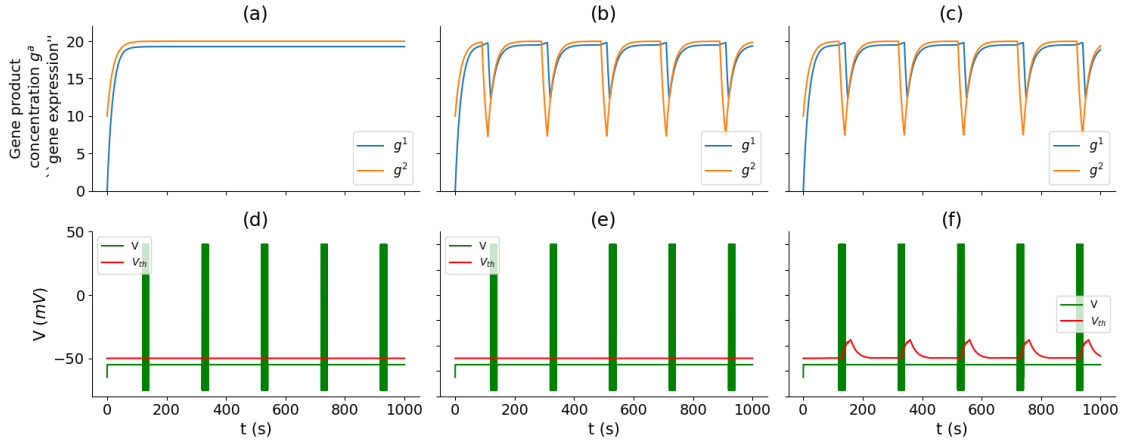


Figure 5: Simulation outcomes for the non-habituation case. (a) Concentration trajectories for the two genes when the neuron and the GRN are uncoupled and the neuron receives the external current in **Figure 1**. **(b)** Concentration trajectories when the GRN depends on neuronal spike rates and the neuron receives the external current in **Figure 1**. **(c)** Concentration trajectories when the GRN is dependent on the neuron ($E^a(r_{stimuli})$) and the neuron is dependent on the GRNs ($V_{th}(r_{base}, r_{stimuli})$) and the neuron receives the external current in **Figure 1**. **(d)** Membrane potential and potential threshold for the neuron corresponding to case (a), when there is no coupling with the GRN. **(e)** Membrane potential and potential threshold corresponding to case (b), for the one-way coupled system (the neuron does not depend on the GRN). **(f)** Membrane potential and potential threshold corresponding to case (c), for the two-way coupled system (V_{th} depends on g^a).

The non-habituation sample results are showed in **Figure 5**. We can see how in **(b)** there is no change on gene concentration stable states with respect to the ones in **(a)**, even if there are some perturbations whenever stimuli arrive. This results in a slight increase on V_{th} after each stimulus when the system is coupled back **(f)**, not being able to incorporate a permanent change, which is how we are modeling learning. Therefore, with our model and the specific GRN parameters chosen for this case we are not able to model habituation learning.

The previous figures show that, as we were expecting, while some GRN parameter configurations capture the habituation behavior on a neuron, some others do not. The next step is to simulate the system for a representative set of GRN configurations over the parameter space and analyse under what circumstances we have a proper model.

3.2 GRN Atlas Analysis

In this section we aim to identify common features of the GRNs that lead to habituation within our model.

We begin by running our algorithm for the 390 000 GRN parameter combinations that were explained in section 2.3 (each parameter combination is the union of a network parameter set or adjacency matrix, a gene external input relationship to the neuronal spike rates and a gene initial concentrations). We obtain, for each of them, a *spike number*, which corresponds to the number of neuronal spikes in our time-window. That will be our feature for the clustering method.

3.2.1 All Possible GRN Configurations

To begin with, we perform a clustering in which all initial concentrations of a same GRN configuration are considered together. In this case, each of our "instances" has 25 features (one spike number for each of the initial condition possibilities). Once the hierarchical clustering has been performed, we yield a dendogram whose two root clusters correspond to a habituation cluster and a non-habituation cluster. The habituation cluster includes all the GRNs with specific parameter configurations that present a habituation behavior similar to that in **Figure 4** for at least one of their possible initial conditions. The non-habituation cluster includes all the GRNs with specific adjacency matrix and E^a that do not present a habituation behavior for any of their possible initial conditions.

To continue, let us try to understand if there are some network topologies that are more prone to yielding habituation with our model. With such a purpose, we generate the two barplots in **Figure 6**. One can appreciate how, indeed, there are some network topologies with a higher count of habituation cases. However, even for those topologies, the number of non-habituation cases wins by a large margin. It is natural to perform the same analysis on GRN external input relationships to the neuron. As it was explained in Section 2.3, there are 8 possible relationship combinations (E^a). In **Figure 7** we appreciate a similar pattern to the one in the barplot of network topologies: there are some external input combinations that have many more habituation instances, though even for those combinations non-habituation is more likely.

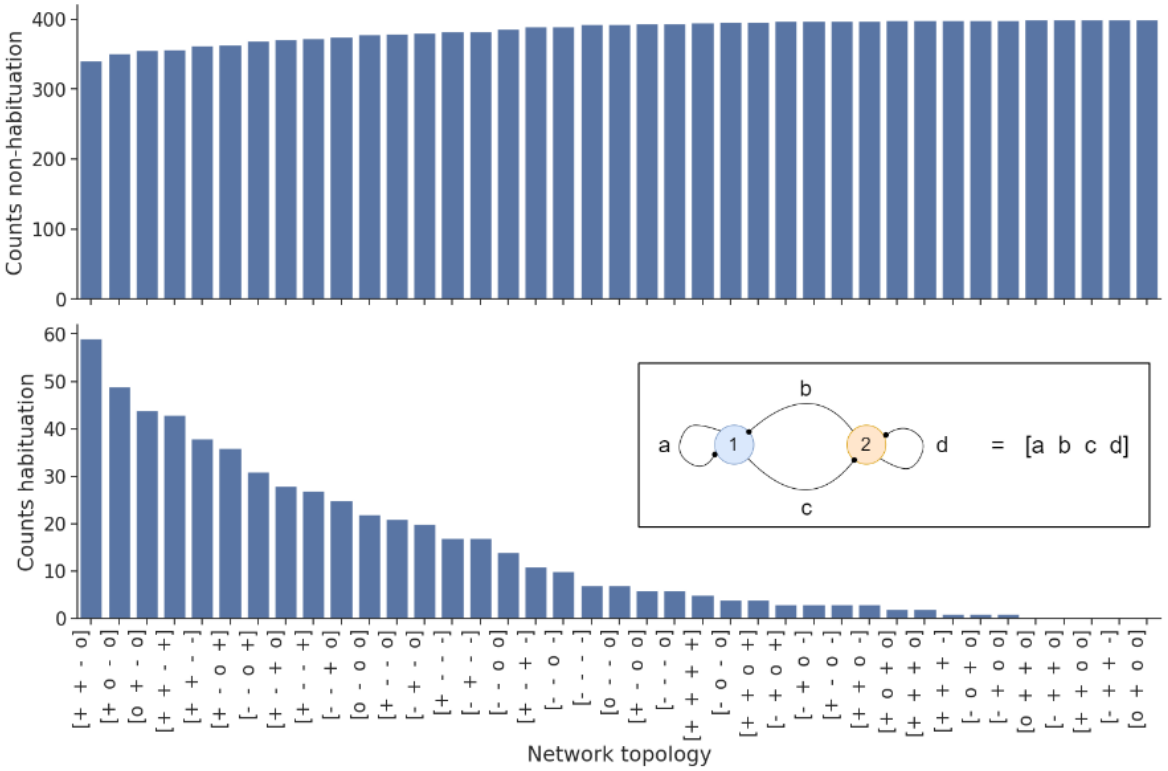


Figure 6: These barplots are counting the number of times one can find a GRN configuration that can not model habituation (top figure) and a GRN configuration that can model habituation (bottom figure) when looking at each possible network topology.

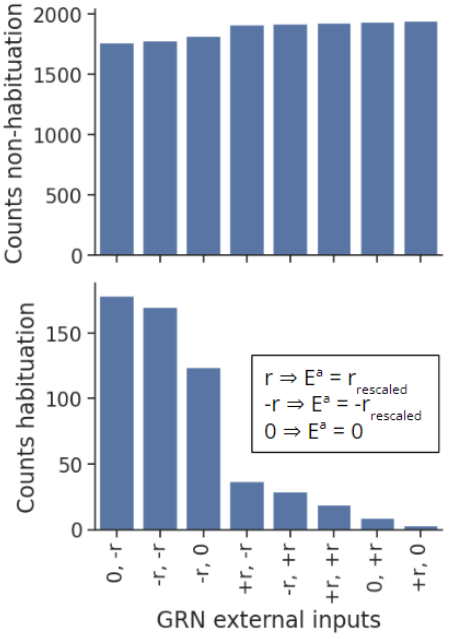


Figure 7: These barplots are counting the number of times one can find a GRN configurations that can not model habituation (top figure) and a GRN configurations that can model habituation (bottom figure) when looking at each possible combination of external inputs to the two genes (*nd* refers to non-dependent and corresponds to $r_{rescaled} = 0$).

The previous results encourage us to look at combinations of networks and GRN external input relationships to the neuron (**Figure 8**).

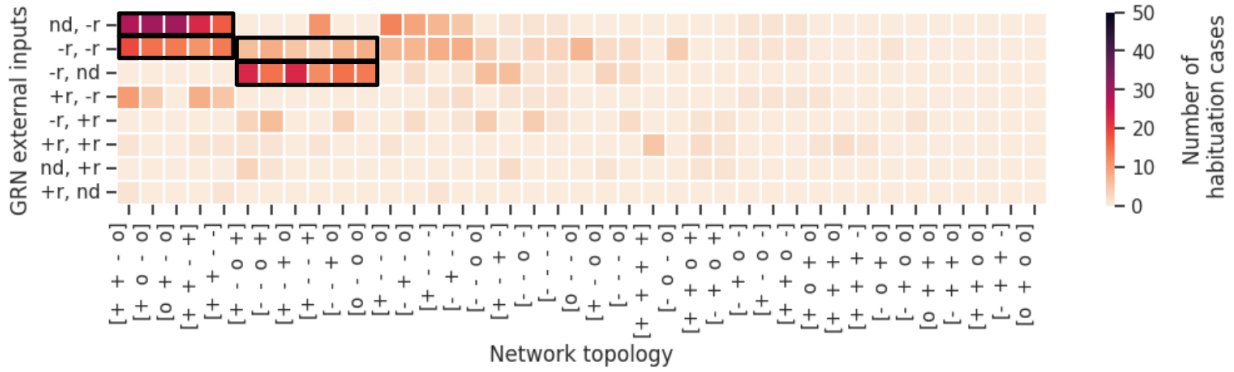


Figure 8: Heatmap representing the number of habituation cases per combination of GRN network topology and external inputs to the genes (GRN configurations).

On the heatmap one can appreciate four main groups in which there is a higher representation of habituation instances. The two first groups (columns 1 to 5 of the heatmap, rows 1 and 2 respectively) correspond to the network topologies in **Figure 9** together with the external inputs $E^1 = 0, \quad E^2 = -r_{rescaled}$ (first group) and $E^1 = -r_{rescaled}, \quad E^2 = -r_{rescaled}$ (second group). The third and fourth group (columns 6 to 12 of the heatmap, rows 2 and 3 respectively) correspond to the network topologies in **Figure 10** together with the external inputs $E^1 = 0, \quad E^2 = -r_{rescaled}$ (third group) and $E^1 = -r_{rescaled}, \quad E^2 = -r_{rescaled}$ (fourth group).

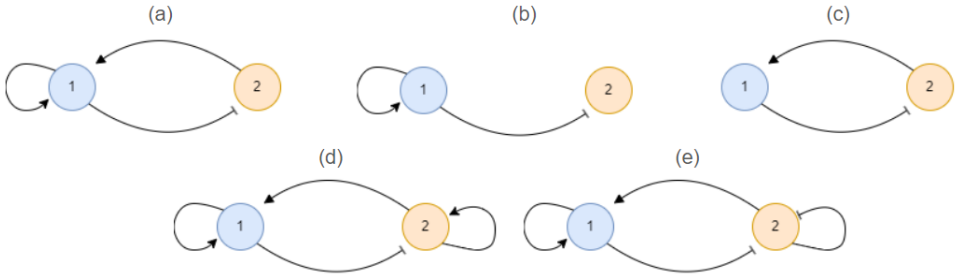


Figure 9: Graphs corresponding to the following GRN network topologies: **(a)** [+ + - o], **(b)** [+ o - o], **(c)** [o + - o], **(d)** [+ + - +] and **(e)** [+ + - -].

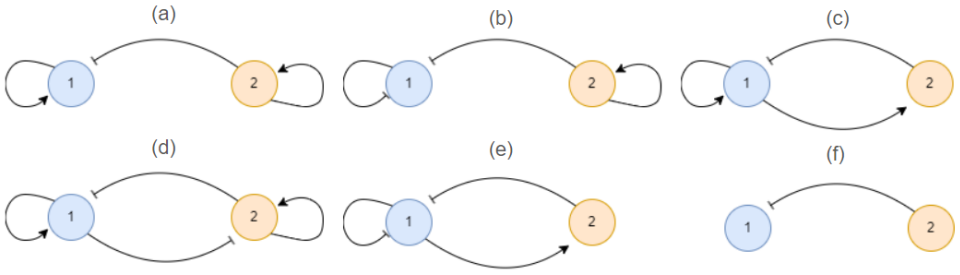


Figure 10: Graphs corresponding to the following GRN network topologies: **(a)** [+ + o +], **(b)** [- - o +], **(c)** [+ - + o], **(d)** [+ - - +], **(e)** [- - + o] and **(f)** [o - o o].

There is a common aspect among all the networks of **Figure 9**: for all of them g^1 inhibits g^2. Between the networks in **Figure 10** there is also a common interaction: g^2 inhibits g^1 in all of them. Moreover, we can appreciate a symmetry between the two first groups and the two last ones: habituation is more probable in GRNs in which one gene inhibits the other and whose genes have either an external input relationship with the neuron such that the inhibitory gene has no dependence on it ($E^{inhibitory} = 0$) and the suppressed gene has $E^{suppressed} = -r_{rescaled}$ or an external input relationship with the neuron such that $E^{inhibitory} = -r_{rescaled}$ and $E^{suppressed} = -r_{rescaled}$.

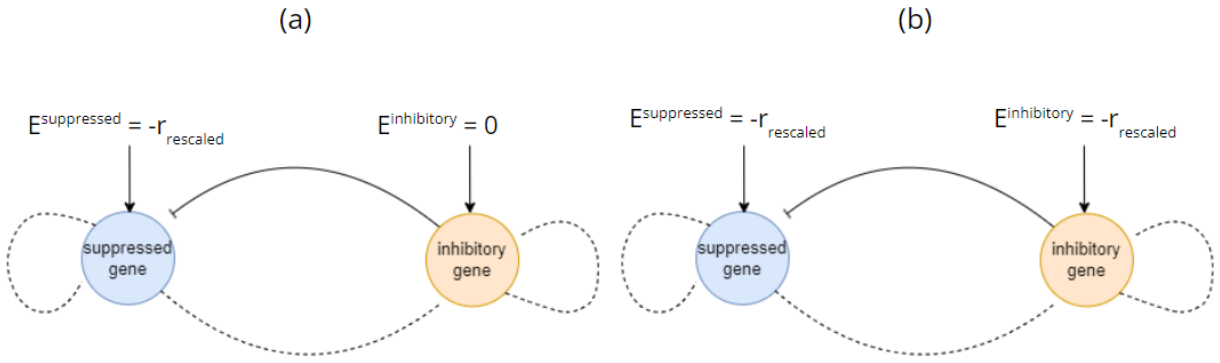


Figure 11: Scheme of the two graphs that seem to be favorable for the modeling of habituation.

3.2.2 GRN Configurations that Present Habituation: Initial Concentrations

But even if a GRN configuration presents habituation for some of its gene initial concentrations, it is not frequent that they present it for all of the possible i.c. In this section we focus only on the fraction of GRNs that present habituation for some i.c., and we try to figure out if there are some preferable initial conditions for the modeling of habituation.

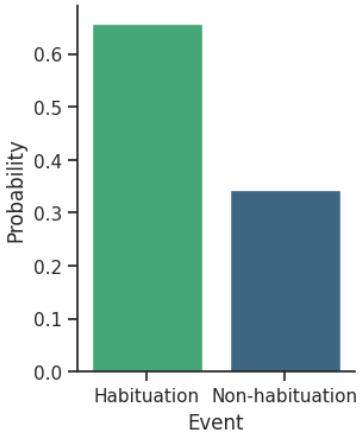


Figure 12: Barplot showing the probability of having a habituation case (GRN with a certain parameter configuration and initial gene concentrations) given the GRN configurations that can present habituation.

To start with, thanks to **Figure 12** we understand that, considering only GRN configurations that do present habituation at some point in their phase portrait, if we give them all the possible sets of initial conditions and run the simulations, about a 65% of the instances present habituation.

For this analysis we have used the hierarchical clustering once again, yielding two principal clusters corresponding to habituatory and non-habituatory instances.

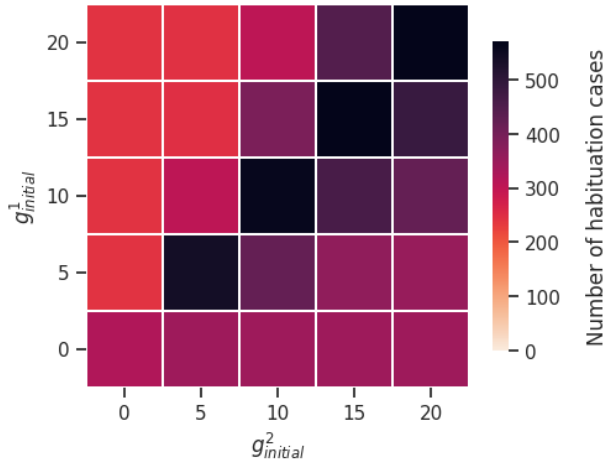


Figure 13: Heatmap that represents a statistical phase portrait of all the GRN configurations that present habituation for some set of initial conditions. Color indicates the number of habituation cases: the larger that number is for a given set of initial conditions, the darker the color of the corresponding square.

It is natural to now wonder whether if, in general, there are some initial concentrations that promote habituation. We can look at **Figure 13** to answer our question: habituation happens almost surely for our habituation-friendly GRN configurations when the two gene initial concentrations are equal. In the rest of the i.c. cases, habituation seems to be more probable for initial concentrations g^1_{ini} and g^2_{ini} that are closer to eachother. For instance, the number of habituation cases for $(g^1_{ini}, g^2_{ini}) = (20, 15)$ is higher than for $(g^1_{ini}, g^2_{ini}) = (20, 10)$, and in both previous cases habituation is more probable than when $(g^1_{ini}, g^2_{ini}) = (20, 5)$. The heatmap is quite symmetric.

Finally, it would be interesting to see if there is a correlation between GRN configurations (GRN with a specific network topology and specific kind of relationship of its genes with the neuron) and the initial conditions for which they do present habituation. In other words: given the GRN configurations of the four habituatory groups that we were able to spot in **Figure 8**, is there a specific set of initial concentrations that allows each group's neuron to habituate? Or is there not a correlation?

By observing **Figure 14** one can notice that, indeed, there is a correlation between the GRN configuration groups and the initial conditions for which they present habituation. While groups 1 and 2 prefer for g^2 to have an equal or grater initial concentration than g^1 so that habituation can happen, groups 3 and 4 go for exactly the opposite. If we look back at **Figures 9** and **10**, we can conclude that, in the ideal setup for the modeling of habituation, it is the suppressed gene in **Figure 11** the one that must, preferably, have a greater or equal initial concentration with respect to the other gene's (the inhibitory one) initial concentration.

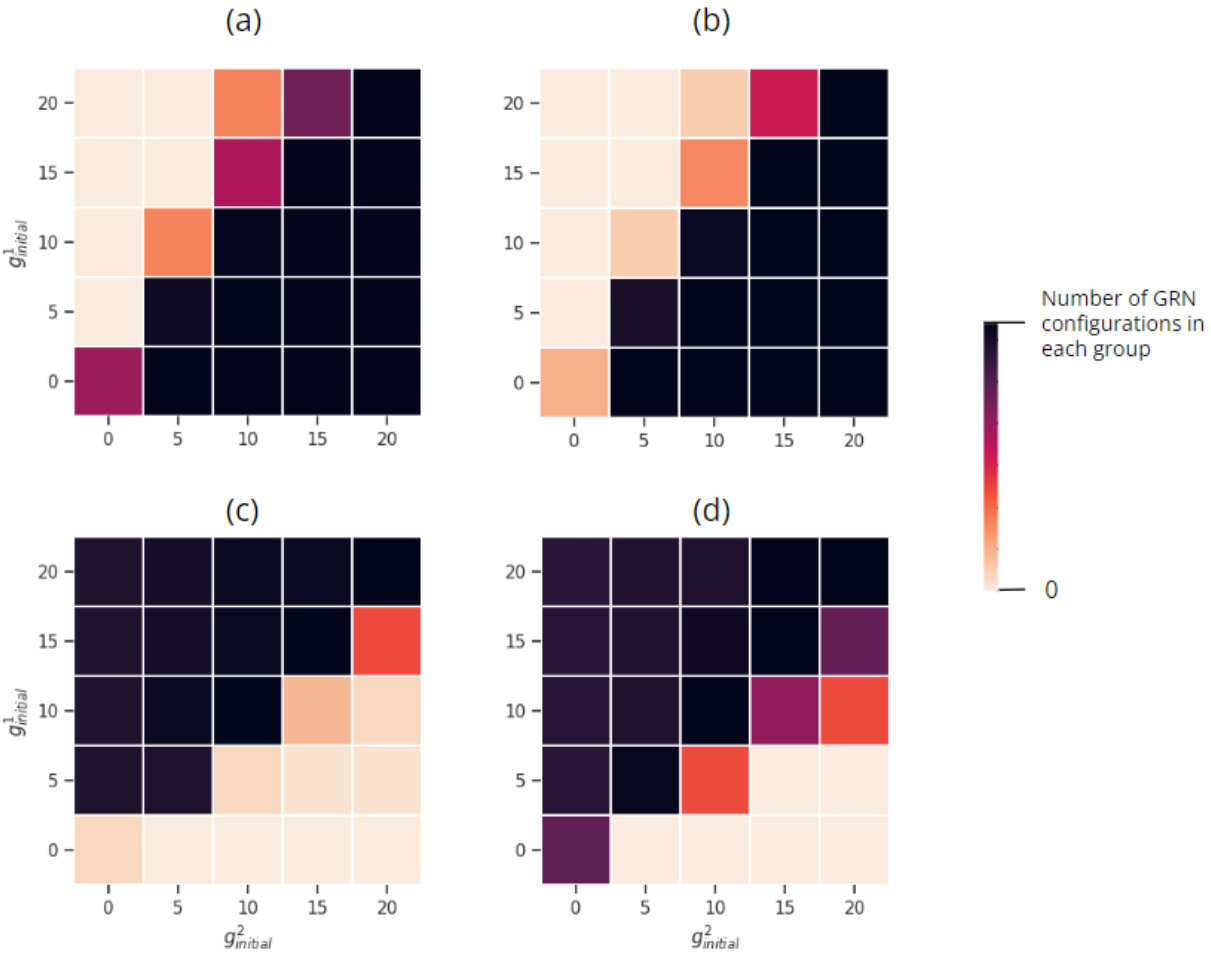


Figure 14: Heatmaps that represents a statistical phase portrait of the four GRN configuration groups (**(a)** group 1, **(b)** group 2, **(c)** group 3 and **(d)** group 4) where the color indicates the number of habituation cases: the larger that number is for a given set of initial conditions, the darker the color of the corresponding square.

4 Discussion and Conclusion

The primary objective of this study was to develop a mathematical model capable of simulating the dynamics of a neuronal system coupled with GRNs within cells, with the aim of understanding how neuronal activity influences gene expression, which in turn facilitates habituation learning in neurons. Our coupled model integrates the LIF model for neuronal activity with a two-gene continuous ODE model for the GRN. The coupling mechanism is designed so that the external inputs to the genes at any given time depend on the neuron's spike rate, while the neuron's membrane potential threshold is influenced by the concentrations of both genes. The model is structured to simulate habituation learning by shifting the stable states of the GRN following an initial stimulus. Through this framework, we determined that for a two-gene GRN to effectively model habituation, it must exhibit specific characteristics: one gene (inhibitory) should suppress the production of the other (suppressed); the suppressed gene should have a specific relationship with the neuron's spike rate such that it is activated during periods of no neuronal activity and inhibited during stimuli; the inhibitory gene's relationship to the neuron offers slightly more flexibility, either mirroring the suppressed gene's relationship or being independent of it; and finally, the initial concentrations should be such that the level of the suppressed gene is equal to or greater than that of the inhibitory gene. With these GRN characteristics in mind, we maximize the probability that our model effectively simulates habituation learning.

Although our model successfully captures the habituation behavior of a neuron, where the response diminishes with repeated stimulation, it relies on several simplifications that limit its direct applicability to real-world scenarios. Firstly, while learning is typically associated with synaptic plasticity (i.e., changes in synaptic strength over time), our model conceptualizes learning as a guided change in the neuron's output spiking dynamics relative to what would be expected in an uncoupled system. This simplified approach allows us to focus on accurately modeling the feedback loop with genes, postponing the complexities of neuronal network interactions until the foundational aspects of the model are firmly established.

Additionally, while our model assumes a two-gene GRN, in reality, genes operate within a complex network of interactions that involve transcription factors (proteins that bind to specific DNA sequences to regulate the transcription of target genes) and various signaling molecules. For simplicity, we have referred to GRNs in terms of gene concentrations throughout this report. Conceptually, our two genes can be considered as"meta-genes", each representing a cluster or network of numerous interacting molecules. Thus, our model captures the interactions between two broader subsystems. This approach to studying molecular networks, inspired by the work of James Sharpe, allows us to analyze complex biological processes in a more tractable manner. In future work, it would be valuable to map our model's GRN to specific components of actual neuronal GRNs involved in learning processes.

Moreover, our model is deterministic and does not account for noise, which is a significant simplification of actual neuronal behavior. In our model, we assume a null spike rate during basal conditions; however, in reality, neurons often exhibit low-frequency spiking during non-stimulated periods, commonly referred to as spontaneous activity. Although we do not expect this simplification to drastically impact the habituation process, it would be valuable to explore the potential dynamic effects that could arise from incorporating stochastic elements into the model.

To further simplify our analysis, we assumed that memory remains constant over time, with no decay. However, this assumption does not accurately reflect real-world conditions, where *dishabit-*

uation typically occurs as time progresses without new stimuli [7]. In future work, we intend to incorporate this dishabituation effect into our model by adjusting the threshold voltage (V_{th}) to better capture the actual dynamics.

Lastly, as discussed earlier in the report, one of the major challenges in coupling neuronal activity, which occurs on a millisecond timescale, with gene regulation, which operates on timescales ranging from seconds to days, is managing the significant difference in timescales. Neuronal activity affects the GRN almost instantaneously, whereas the resulting changes in gene concentrations impact neuronal behavior only after a delay. To address this, we have currently implemented a 10-second delay in the algorithm to simulate the effect of genes on neurons. However, this delay is minimal and arbitrarily chosen. A valuable next step would be to systematically tune this parameter and analyze its impact on the simulation outcomes. Additionally, it would be beneficial to incorporate this timescale difference into the mathematical model itself, potentially by using delayed differential equations in future work.

In addition to the future work suggested by the previously discussed simplifications, there are several other possibilities we are eager to explore within this framework. One intriguing direction is to investigate other types of learning beyond habituation. A straightforward extension would be to model sensitization, which is nearly the opposite of habituation: in this case, the more frequent the spikes, the stronger the response, as the stimuli are perceived as increasingly threatening. Another compelling challenge would be to validate our models using empirical data and refine them to more accurately reproduce observed phenomena. This direction is particularly exciting given that advancements in single-cell technologies now allow us to measure mRNA levels (gene expression) at a genome-wide scale, which was a key motivation for initiating this study.

In conclusion, while our model is still in its early stages, it establishes a promising foundation for exploring the intricate relationship between neuronal activity and GRNs in the context of habituation learning. Although we have made several simplifying assumptions to make the model tractable, these simplifications have opened up numerous possibilities for future research. Further development and refinement are necessary to assess whether the model's scope can be broadened or if it remains limited. Future work should focus on incorporating more realistic elements, such as noise, and validating the model with empirical data from advanced technologies. With continued research, this model has the potential to enhance our understanding of molecular-level learning processes, offering valuable insights for the communities of theoretical neuroscience and computational biology.

References

[1] S. W. Flavell and M. E. Greenberg, "Signaling mechanisms linking neuronal activity to gene expression and plasticity of the nervous system," *Annual Review of Neuroscience*, vol. 31, pp. 563–590, 2008.

[2] A. E. West and M. E. Greenberg, "Neuronal activity-regulated gene transcription in synapse development and cognitive function," *Cold Spring Harbor Perspectives in Biology*, vol. 3, no. 6, p. a005744, 2011.

[3] P. L. Greer and M. E. Greenberg, "From synapse to nucleus: calcium-dependent gene transcription in the control of synapse development and function," *Neuron*, vol. 59, no. 6, pp. 846–860, 2008.

[4] H. Bading, "Nuclear calcium signalling in the regulation of brain function," *Nature Reviews Neuroscience*, vol. 14, no. 9, pp. 593–608, 2013.

[5] H. Bading, "Nuclear calcium signalling in the regulation of brain function," *Nature Reviews Neuroscience*, vol. 14, pp. 593–608, Sept. 2013.

[6] I. B. Zovkic, M. C. Guzman-Karlsson, and J. D. Sweatt, "Epigenetic regulation of memory formation and maintenance," *Learning & Memory*, vol. 20, pp. 61–74, Jan. 2013. Epub 2013 Jan 15.

[7] C. H. Rankin, T. Abrams, R. J. Barry, S. Bhatnagar, D. F. Clayton, J. Colombo, G. Coppola, M. A. Geyer, D. L. Glanzman, S. Marsland, F. K. McSweeney, D. A. Wilson, C. F. Wu, and R. F. Thompson, "Habituation revisited: an updated and revised description of the behavioral characteristics of habituation," *Neurobiology of Learning and Memory*, vol. 92, pp. 135–138, September 2009.

[8] Y. Shen, S. Dasgupta, and S. Navlakha, "Habituation as a neural algorithm for online odor discrimination," *Proceedings of the National Academy of Sciences*, vol. 117, no. 22, pp. 12402–12410, 2020.

[9] D. L. Glanzman, "Habituation in aplysia: the cheshire cat of neurobiology," *Neurobiology of Learning and Memory*, vol. 92, pp. 147–154, September 2009.

[10] W. Cho, U. Heberlein, and F. W. Wolf, "Habituation of an odorant-induced startle response in drosophila," *Genes, Brain and Behavior*, vol. 3, pp. 127–137, June 2004.

[11] M. P. Leussis and V. J. Bolivar, "Habituation in rodents: a review of behavior, neurobiology, and genetics," *Neuroscience Biobehavioral Reviews*, vol. 30, no. 7, pp. 1045–1064, 2006.

[12] K. M. Tyssowski, N. R. DeStefino, J.-H. Cho, C. J. Dunn, R. G. Poston, C. E. Carty, R. D. Jones, S. M. Chang, P. Romeo, M. K. Wurzelmann, J. M. Ward, M. L. Andermann, R. N. Saha, S. M. Dudek, and J. M. Gray, "Different neuronal activity patterns induce different gene expression programs," *Neuron*, vol. 98, no. 3, pp. 530–546.e11, 2018.

[13] V. P. Zhdanov, "A neuron model including gene expression: Bistability, long-term memory, etc.," *Neural Processing Letters*, vol. 39, no. 3, pp. 285–296, 2014.

[14] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

[15] G. Karlebach, R. Shamir, D. Cicin-Sain, M. Ashyraliyev, and J. Jaeger, "Modelling and analysis of gene regulatory networks," *Nat Rev Mol Cell Biol*, 10 2008.

[16] J. Cotterell and J. Sharpe, "An atlas of gene regulatory networks reveals multiple three-gene mechanisms for interpreting morphogen gradients," *Molecular Systems Biology*, vol. 6, no. 1, p. 425, 2010.

[17] M. B. Kennedy, "Synaptic signaling in learning and memory," *Cold Spring Harbor Perspectives in Biology*, vol. 8, no. 2, p. a016824, 2013. Review.

[18] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. New York: Psychology Press, 1st ed., 2002.

[19] P. Dayan and L. F. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press, 2005.

[20] N. Academy, "Nma computational neuroscience course."

[21] H. de Jong, "Modeling and simulation of genetic regulatory systems: A literature review," *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002. PMID: 11911796.

[22] A. Polynikis, S. Hogan, and M. di Bernardo, "Comparing different ode modelling approaches for gene regulatory networks," *Journal of Theoretical Biology*, vol. 261, no. 4, pp. 511–530, 2009.

[23] A. Crombach, K. R. Wotton, D. Cicin-Sain, M. Ashyraliyev, and J. Jaeger, "Efficient reverse-engineering of a developmental gene regulatory network," *PLOS Computational Biology*, vol. 8, pp. 1–21, 07 2012.

[24] E. Mjolsness, D. H. Sharp, and J. Reinitz, "A connectionist model of development," *Journal of Theoretical Biology*, vol. 152, no. 4, pp. 429–453, 1991.

[25] J. Helton and F. Davis, "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems," *Reliability Engineering System Safety*, vol. 81, no. 1, pp. 23–69, 2003.

[26] C. Song and R. Kawai, "Monte carlo and variance reduction methods for structural reliability analysis: A comprehensive review," *Probabilistic Engineering Mechanics*, vol. 73, p. 103479, 2023.