# Hidden complexity in biologically inspired neural networks : emergent properties and dynamical behavior

Jules Bouté

▶ **To cite this version:**

Jules Bouté. Hidden complexity in biologically inspired neural networks : emergent properties and dynamical behavior. Neuroscience. Université Paris-Saclay, 2023. English. NNT : 2023UPASL074 . tel-04484179

**HAL Id: tel-04484179**
**https://theses.hal.science/tel-04484179**

Submitted on 29 Feb 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

université PARIS-SACLAY

# Hidden complexity in biologically inspired neural networks : emergent properties and dynamical behavior

*La complexité cachée dans les réseaux neuronaux d'inspiration biologique : propriétés émergentes et comportements dynamiques*

**Thèse de doctorat de l'université Paris-Saclay**

École doctorale n° 568 Signalisations et réseaux intégratifs en biologie (Biosigne)
Spécialité de doctorat : Neurosciences
Graduate School : Life Sciences and Health
Référent : Faculté de médecine

Thèse préparée à l'***Institut des neurosciences Paris-Saclay (Université Paris-Saclay, CNRS)***, sous la direction de **Alain DESTEXHE**, Directeur de Recherche CNRS

**Thèse soutenue à Paris-Saclay, le 19 septembre 2023, par**

**Jules Bouté**

## Composition du Jury

Membres du jury avec voix délibérative

| | |
|---|---|
| **Antoine CHAILLET**<br>Directeur de recherche, L2S CentraleSupelec | Président |
| **Sacha VAN ALBADA**<br>Directrice de recherche, University of Cologne | Rapporteur & Examinatrice |
| **Serafim RODRIGUES**<br>Directeur de recherche, Basque Center of Applied Mathematics | Rapporteur & Examinateur |
| **Bruno CESSAC**<br>Directeur de recherche, NRIA Biovision Lab | Examinateur |

Thèse de doctorat
NNT : 2023UPASL074

université PARIS-SACLAY

ÉCOLE DOCTORALE
Signalisations et réseaux intégratifs en biologie (BIOSIGNE)

**Titre :** La complexité cachée dans les réseaux neuronaux d'inspiration biologique : propriétés émergentes et comportements dynamiques

**Mots clés :** neuroscience computationnelle, modelisation, systèmes dynamiques, exposants de Lyapunov, dissipation

**Résumé :** Dans cette thèse, nous étudions la dynamique neuronale pendant l'éveil attentif, en se concentrant spécifiquement sur le type de dynamique "asynchrone et irrégulière" (AI) observée dans le cortex cérébral. On vise ici à mieux comprendre les comportements des réseaux neuronaux, en mettant l'accent sur deux modèles : le modèle adaptatif exponentiel (AdEx) "integrate and fire", et le modèle Hodgkin-Huxley (HH), avec un accent particulier sur le modèle AdEx.

La thèse met en évidence des différences intrigantes entre des réseaux neuronaux apparemment similaires.

Dans une première partie, nous montrons la manière dont les réseaux neuronaux répondent aux stimuli externes, en particulier dans le contexte de l'épilepsie, en examinant pourquoi certains systèmes contrôlent et "tuent" l'entrée paroxystique alors que d'autres la reproduisent et la propagent.

Dans une deuxième partie, la thèse explore la manière dont les réseaux de neurones AdEx ou HH faiblement connectés peuvent présenter des états AI de deux manières distinctes : des états "auto-entretenus" ou des états "pilotés" nécessitant une entrée externe (Driven).

Dans une troisième partie, nous présenterons un autre outil de la physique statistique moins usité : la dissipation. Nous concentrerons alors notre travail sur les analyses des modèles et des différents réseaux permises par cet outil.

Pour élucider ces différences, une combinaison de techniques classiques de traitement des signaux, d'analyse structurelle de la configuration du réseau et d'outils de systèmes dynamiques - notamment les exposants de Lyapounov et la dissipation du système - est employée. L'étude que nous avons réalisée utilise également des modèles de champ moyen des réseaux AdEx pour comprendre comment les variantes auto-entretenues et pilotées se manifestent à un niveau global.

L'exploration de ces propriétés et variations améliore notre compréhension des propriétés émergentes complexes et des comportements dynamiques distincts qui existent au sein de réseaux neuronaux biologiques apparemment similaires.

**Title :**Hidden complexity in biologically inspired neural networks : emergent properties and dynamical behavior

**Keywords :** computational neuroscience, modelization, dynamical systems, Lyapunov exponents, dissipation

This thesis investigates the neural dynamics during attentive wakefulness, specifically focusing on the "asynchronous and irregular" (AI) type of dynamics observed in the cerebral cortex. The research aims to gain a deeper understanding of neural network behaviors, with a primary emphasis on two models: the Adaptive Exponential (AdEx) integrate and fire model, and the Hodgkin-Huxley (HH) model, with a particular focus on the AdEx model.

The thesis uncovers intriguing differences among seemingly similar neural networks.

In a first part, it will shed light on how neural networks respond to external stimuli, particularly in the context of epilepsy, examining why certain systems control and kill the paroxysmal input while others reproduce and propagate it.

In a second part, the thesis explores how sparsely connected networks of AdEx or HH neurons can exhibit AI states in two distinct ways: self-sustained AI states, or AI states requiring external input (Driven).

In a third part, we will introduce the Dissipation and focus our analysis of models and different networks around this tool.

To unravel those differences, a combination of classical signal processing techniques, structural analysis of the configuration of the network, and dynamical systems tools - including Lyapunov exponents and system dissipation - are employed. The investigation also makes use of mean-field models of AdEx networks to comprehend how self-sustained and driven variants manifest at a global level.
Exploring these properties and variations enhances our understanding of the complex emergent properties and distinct dynamical behaviors that exist within apparently similar biological neural networks.

# Synthèse

Cette thèse vise à étudier la compléxité des réseaux neuronaux, un assemblage de modèles de neurones uniques compréhensibles (et aisément représentables et visualisable, bien qu'apportant déjà une dynamique riche) générant des propriétés émergentes complexe et des dynamiques à très hautes dimensions impossible à représenter directement. Il est donc nécessaire de produire des outils spécifiques pour analyser ces systèmes, outils qui seront souvent en rapport plus ou moins direct avec la moyenne. Or, nous allons montrer au cours de cette thèse que des réseaux avec des comportements moyens similaires - c'est à dire par exemple avec une "fréquence de tir" calculée en prenant la moyenne de la fréquence de tir de chaque neurone - qu'on pourrait donc avoir tendance à qualifié de dynamiquement similaire, peuvent en fait se révélé être être très différent dans la pratique si on utilise d'autres outils plus adaptés.
Pour ce faire, nous allons piocher dans l'arsenal des systèmes dynamiques, en utilisant différentes méthodes d'analyses, et en particulier le « coarse graining », les exposants de lyapunov et la dissipation.

Étudier exhaustivement tous les modèles, bien que passionnant, n'aurait pas été possible faute de temps. En guise de cas d'étude, et parce qu'ils sont assez répandus tout en couvrant des usages et des niveaux de complexités différents, nous nous concentrons sur deux modèles de neurones uniques : le modèle adaptatif exponentiel (AdEx) d'intégration et tir, et le modèle Hodgkin-Huxley (HH), avec un accent particulier sur le modèle AdEx. Dans chaque simulation produite au cours de ce travail, les réseaux sont construits à partir de 10 000 neurones connectés aléatoirement avec une faible probabilité (5%). Dans un même réseau, chaque neurone suis le même modèle (AdEx ou HH) et nous produisons à chaque fois une population inhibitrice et une population excitatrice en changeant les paramètres spécifiques des neurones. Tous neurones excitateurs sont identiques entre eux, de même que les inhibiteurs, et les différences émergeront avec les différentes connexions. Plus spécifiquement, cette thèse étudie la dynamique neuronale pendant l'éveil attentif, en se concentrant spécifiquement sur le type de dynamique "asynchrone et irrégulière" (AI) observée dans le cortex cérébral, et nous utilisons donc des paramètres pertinent pour reproduire les données in vivo associées.

La thèse s'ouvre sur un chapitre introductif, avant de suivre trois chapitres abordant chacun des angles spécifiques de différences entre les réseaux, puis de finir par une conclusion synthétisant l'ensemble.
L'introduction explique plus en avant les bases de neurosciences computationnelles utilisées dans la thèse et nécéssaire à la compréhension des simulations, avant d'expliquer le choix de l'outil d'analyse utilisé : les systèmes dynamiques .
Le chapitre deux montre comment le hasard - obtenu par des connectivités ou des "bruits poissoniens" différents - suffit à avoir des comportements dynamiques complètement différents malgré des réseaux identiques, mise à part l'aléatoire spécifiés.
Le chapitre trois illustre la différence entre réseaux dits « driven », ou « entraînés » et réseaux dits « self-sustained », ou « auto-entretenus », que nous expliciterons, au travers des exposants de Lyapunov.
Le chapitre quatre, ensuite, montre comment la dissipation, un outil peu utilisé dans l'étude des réseaux de neurones, illustres des différences entre modèles de neurones et entre types de réseaux, en insistant particulièrement sur la réponse à une perturbation.
Nous allons à présent détailler les différentes parties.

L'introduction commence par souligner l'importance de la modélisation de la réalité et la nécessité d'un examen attentif des représentations.
Nous nous penchons ensuite sur l'analyse de réseaux neuronaux biologiquement réalistes, détaillant la modélisation de neurones uniques et le détail de la création des différents réseaux, en mettant l'accent sur les réseaux entraînés et auto-entretenus. Les premiers ont besoin d'un d'être constamment excités par une source extérieurs (sans quoi ils n'ont plus d'activité), quand les seconds ont juste besoin d'un démarrage initial qui peut ensuite être coupé.
L'approche analytique choisie est ensuite introduite en tant que théorie des systèmes dynamiques, présentant des concepts clés et insistant sur les exposants de Lyapunov, un outil classique permettant de mesurer la stabilité d'un système, et la dissipation, un outil plus complexe représentant la vitesse à laquelle le système converge vers un sous-espace spécifique.

Le chapitre deux reprend un article publié en 2022(Depannemaecker, Carlu, Bouté, & Destexhe, 2022).
Cette étude computationnelle examine comment l'activité épileptique envahit le tissu cérébral normal et montre le rôle spécifique de la population inhibitrice, ainsi que ses aspects dynamiques et structurels, à l'aide de trois réseaux neuronaux différents. Nous mettons en lumière l'importance des aspects structurels

et dynamiques pour déterminer si l'activité épileptique envahit ou non le réseau. Nous constatons également que, malgré ces déterminants, une partie de la raison de l'effet reste aléatoire, ce qui laisse présager une dynamique complexe que nous ne maîtrisons pas totalement. Nous montrons enfin l'existence d'une fenêtre temporelle spécifique favorable à l'inversion de la propagation des crises par des stimuli appropriés.

Le chapitre trois reprend un article pré-publié(Bouté & Destexhe, 2023).
Cette partie a pour but d'élucider les différences entre deux systèmes apparemment similaires, à savoir les réseaux entraînés et les réseaux auto-entretenus présentés précédemment. Ici, les différents réseaux ont une activité moyenne similaire, et pourraient donc paraître de même nature si seule cette activité (décharge moyenne) était prise en compte.
Cependant, nous avons montré que, pour des réseaux AdEx (mais pas pour les réseaux HH), ce dernier avait un premier exposant de Lyapunov d'un ordre de grandeur plus élevé que le premier, ce qui se traduit par une tendance à avoir une réponse plus élevée aux perturbations. Nous avons également validé les résultats de ces réseaux sur un modèle de champ moyen. Ces résultats ne sont cependant pas valides pour les réseaux HH, ce qui nous a amenés à creuser les questions des différences entre ces modèles avec d'autres outils.

Le chapitre quatre n'est pas encore publié, et reprend différents résultats obtenus en étudiant la dissipation des réseaux.
Comme indiqué précédemment, nous utilisons ici la dissipation (et comme précédemment, les réseaux ont, autant que possible, une activité moyenne similaire). Cette mesure nous a permis de produire des observations de l'échelle du neurone à l'échelle du réseau, et a révélé de nouvelles similitudes et différences entre les neurones AdEx et HH, d'une part, et les réseaux entraînés et auto-entretenus, d'autre part.
Nous avons d'abord vérifié que la dissipation des réseaux prédisait bien la vitesse de convergence d'un réseau vers une activité stable.
Ensuite, nous avons montré que la corrélation entre le taux de décharge moyen et la dissipation moyenne prédisait le type de réseau – entraînés ou auto-entretenus – indépendamment du modèle, nous permettant donc d'obtenir des résultats plus robustes de ceux de la partie précédente.
Enfin, une attention particulière a été portée à la réactivité des réseaux, et nous avons montrer que la dissipation des réseaux AdEx (mais pas HH cette fois) prédisait mieux la réponse du réseau à une perturbation que la cadence de tir moyenne.

Pour conclure, et comme énoncé précédemment, cette thèse explore les différences entre des réseaux apparemment similaires, en se concentrant sur les modèles AdEx et HH.

Les réseaux AdEx présentent des dichotomies telles que la bistabilité et des différences dynamiques entre les réseaux entraînés et les réseaux auto-entretenus, ce qui suggère une meilleure capacité à imité des phénomènes avec des bifurcations et des comportements variés et bien catégorisés. Les réseaux HH présentent quant à eux une continuité avec moins de variabilité, étant donc plus appropriés pour les phénomènes continues, sans frontières strictes, mais étant aussi plus robuste face aux changements de paramètres.

Les dynamiques diffèrent considérablement entre les réseaux entraînés et les réseaux auto-entretenus, même avec des sources excitatrices externes similaires, en particulier dans les réseaux AdEx, comme on peut le voir en étudiant leur premier exposant de Lyapunov. Il est important de noter que l'étude de la dissipation a mis en avant des différences dynamiques similaires pour les réseaux HH, permettant d'envisager une généralisation de ces différences si l'on use des outils appropriés.

Il est important de noter que l'étude de la dissipation a mis en avant des différences dynamiques similaires pour les réseaux HH, permettant d'envisager une généralisation de ces différences si l'on use des outils appropriés.

Dans cette thèse, nous utilisons la dissipation, un outil peu utilisé dans les neurosciences computationnelles mais néanmoins puissant, offrant une perspective différente et informative par rapport aux plus classiques exposants de Lyapunov. Cette thèse plaide donc en faveur d'une utilisation plus importante de la dissipation pour mieux comprendre la dynamique des réseaux de neurones.

Enfin, le principal résultat sur lequel nous voulons insister est la nécessité d'une analyse plus systématique des réseaux neuronaux que nous utilisons. Ces objets constitués d'un modèle de neurones uniques - que nous comprenons en grande partie - laissent émerger des dynamiques complexes que nous sommes encore loin de maîtriser complètement, et il est probable qu'un grand nombre de réseaux aux activités apparemment similaires aient en fait une dynamique fondamentalement différente, qui nous donnerait

alors des réponses différentes aux questions que nous nous posons. Une meilleure compréhension de la dynamique de ces réseaux, de leurs spécificités, de leurs différences ou de leurs similitudes, et de leur raison d'être, nous aiderait à utiliser ces outils avec plus de précision et, en fin de compte, à mieux comprendre le cerveau.

# Remerciements

Je tiens tout d'abord à remercier Alain Destexhe à sa juste valeur pour sa supervision et ses conseils, merci Alain, je n'aurais pas pu faire cette thèse sans toi. Je souhaite ensuite remercier les membres du jury pour leur disponibilité et leurs remarques pertinentes et stimulantes. Je remercie ensuite Thaly pour son financement, et le Human Brain Project pour m'avoir permis de terminer ma thèse aussi sereinement que possible.

Je remercie ensuite toutes les personnes qui m'ont supporté au labo. Mallory et Damien, il est clair que je n'aurais pas pu faire cette thèse sans les discussions qu'on a pu avoir, les idées qui ont émergé de ces dernières, et les conseils que vous avez pu me donner. Coralie, Duda, Thomas, Tom, Zéli, merci pour les pauses-café (ou thé ?) et les discussions (parfois scientifiques) qu'on a pu avoir qui ont rendu ma vie sociale au labo des plus plaisante ! Merci aussi au reste de l'équipe pour les discussions intéressantes que j'ai pu avoir avec vous.

Enfin, parce qu'une thèse est difficile à tenir (et c'est d'ailleurs un problème, mais les remerciements ne sont sans doute pas le meilleur endroit pour en parler), je remercie mes ami.e.s et ma famille pour m'avoir permis de la supporter et de la mener jusqu'au bout, seul, je n'y serais pas parvenu.
Je remercie les Coguys, pour votre aide quand j'en avais besoin et tous les bons moments le reste du temps. Anne-So et Morgan pour le tout premier Summer CogniSchool ainsi que, Morgan, pour m'avoir prété des chaussures quand j'en avais besoin [1]( elles sont désormais rendues). Camille pour sa joie de vivre et 5tob1. Carole et Théo pour les parties de TM pendant les couvre-feux. Charlélie et Thoto pour les très, très nombreuses fois où j'ai squatté votre appart pour tout un tas de raison différente, au point que je me demande si je ne devrais pas commencer à payer des charges. Joffrey, pour m'avoir appris plein de trucs cools en philo, en muscu et pour un super weekend avec des annemasse de fromage. Julie pour avoir toujours été supportive même quand elle n'avait pas de temps pour elle, et pour m'avoir fait découvrir Nantes. Kevin pour les magnifiques week-ends en Suisse et pour tout ce qu'il apporte (et pas qu'en fromage/chocolat). Morgie et Pierre, pour des week-ends en Belgique cette fois, et pour la prochaine dégustation de bière. J'aurais certainement mille choses à dire de plus, mais on va s'arrêter là. Je remercie aussi JB pour avoir pris le temps de faire des bières et d'en boire, même quand il n'avait pas une seconde à lui. Loutre, pour avoir été d'une aide inestimable quand j'avais des blocages, aussi bien scientifiques qu'administratifs. Tam, pour m'avoir montré que faire une thèse en militant était périlleux mais néanmoins possible.
Je remercie aussi, bien entendu, ma famille pour son support permanent. Ma mère, Nathalie pour son écoute, sa curiosité, et sa tentative de comprendre ce que je faisais même quand ça n'était pas vraiment clair pour moi. Ma sœur, Mégane, pour m'avoir soutenu même en ne comprenant pas grand-chose à ce que je faisais, et en m'aidant quand j'en avais besoin. Mon père, Frédéric, pour avoir cru en moi, sans qui cette thèse n'aurait pas existé. Et toute ma famille pour sa chaleur précieuse et ses encouragements.

En plus des gens, je remercie en vrac les mix youtube de musiques de jeux vidéos, sans qui je n'aurais jamais pu resté concentré aussi longtemps, les raclettes, les dessins-animés inclusifs, le 13 décembre, les discussions pendant les pauses cafés, les livres et jeux vidéos qui m'ont aidé à me déconnecté, et mon chat qui m'a tenu compagnie pendant de longs moments de télétravail et de rédaction.

Enfin, je tiens à remercier Luiza. Tu as pu voir toute ma thèse, du début à la fin, et tu as été à la fois mon plus grand soutien - même à distance - quand les choses n'allait pas, et mon plus grand bonheur quand les choses allaient bien. Merci pour tout.

[1] Morgan Beaurenaut. Social cognition under threat anxiety-inducing contexts. Cognitive Sciences. Université Paris sciences et lettres, 2021. English. NNT : 2021UPSLE012. tel-03726520

## Abbreviations

- **AI** : Asynchronous Irregular (macroscopic state of the network)
- **AdEx** : Adaptative Expontential Integrate and Fire (model of single neuron)
- **CAdEx** : Conductance-based Adaptive Exponential integrate-and-fire (model of single neuron)
- **EEG** : Electroencephalography
- **FLE** : First Lyapunov Exponent
- **FR** : Firing Rate(s)
- **FS** : Fast Spiking neurons (equivalent to inhibitory neurons)
- **HH** : Hodgkin Huxley (model of single neuron)
- **LE** : Lyapunov Exponent(s)
- **LIF** : Leaky Integrate and Fire (model of single neuron)
- **RS** : Regular Spiking neurons (equivalent to excitatory neurons)
- **UD** : Up and Down (macroscopic state of the network)
- **Vm** : Membrane potential

# Contents

# Chapter 1 : Introduction

# 1 What is it all about ?

Neural networks are made of single neurons, connected together in various ways to form a network. While this definition is obvious, it leads to a lot of interesting and deep questions. To explore them, we will go back from single neurons to how they are assembled to create networks, to the study of said networks as an emergent whole.

But first, let us take a step back on the concept of models and modelling, specifically from (Frigg & Hartmann, 2020). Models are often used as a way to represent reality in an easier and manipulable manner, meaning a first definitions of models in science could be "simpler representation of complicated phenomenon". But why is it so important to simplify reality ? Because reality itself is to complex, to intertwined for us to understand it as a whole. We need to break it to smaller pieces, partially false, but at least meaningful for our brain.

Ontologically wise, models can take various forms, from physical representation (e.g. small solar system toy model), to abstract/fictional objects (think of Bohr model of atom), to description and, more specifically, mathematical descriptions with equations, which is what we focus on here. Those models aim to describe the reality accurately within a certain range. As said before, it is still a simplified representation of reality, as we conceptualize an object as if it had some intrinsic reality outside of the rest of the possible interactions it can have, but for that range and that conception, the object is represented well.

We can think here of the model of neurons : the goal is to study the neuron themself, it can be a 3D object with lots of connections and gates modeled, and lots of electrophysiologic interactions. If the goal is to represent the membrane potential of a neuron, it is possible to model things at the scale of the molecules. That would be impossible if the goal was to represent the whole neuron, let alone a network of them. But the limitations are not only on the scale. For example, neurons could also need models of various astrocytes to have a complete behavior, and probably of many other objects, some of which could still be unknown to us. On the other hand, if we want to model the effect on firing rate and the communication between neurons, then all that precision on the morphology of the neuron or the precise behavior of the gates is no longer needed, and would be impossible to use at a bigger scale. Still, those models, said of point neurons, are as valid as the previous ones, as the goal of models is not to have a perfect representation of reality, but to have a useful representation of it.

This is why models are so important epistemologically speaking. As it is well known, George Box once said "All models are false, some are useful". All models we use, including in this work, are always false in the sense that they are not a perfect replication of reality. But they are still useful, because they allow us to learn about them, to explore them, and to create knowledge for them. Knowledge that, and it is important, is specifically valid for the model, was possible to obtain because it is a simplified representation of reality.
It is not certain if the knowledge we generate from models represent reality well, and this is an important debate in epistemology (Chalmers, 2013; Frigg & Hartmann, 2020). But whether it does represent reality directly or not, it is often knowledge that is operational, that can be used directly to influence reality in a known way that is interesting to us. Yes, all models are wrong. They all have limitations, and are only valid (in the sense that it is meaningful to use them) in a given range, a range that is often not well known. This is an important point that was the beginning of this work : models are useful as long as they are used correctly, in the range we know they are working in.

But a problem arose : some models are very complicated, especially when they are a construction made from different models, or even similar models interacting together. This is often (albeit, not always) the case for neural networks : we have an established, already complex single neuron model that is well studied and understood, and that has known limitations. And then, we put those modeled single neurons together, creating emergent complex, complicated phenomena that actually become too big (when there are enough neurons) to conceptualize. We created a model we can use, but no longer fully understand and manipulate with ease, as would be the case, for example, of some mean-field models of a network : although very complex and able to produce meaningful behaviors, they are way simpler to understand, to conceptualize and therefore to properly use than networks of single neurons models.

The main goal of this thesis will be to show that apparently similar networks - meaning they are both representations from the same object and, while it is known they are different, are functionally used as interchangeable - produce actually different behavior.
As the subject would be too large to analyze entirely, we focus on some specific examples, both being

made from changing the single neuron's precise descriptions and having influence on the whole network.

We first have the difference between models of single neurons that produce similar activities once put in a network, and then the differences between a dichotomy of networks that appear while changing the parameters of the single neuron models : Driven and Self-sustained networks.

To be able to study those cases, we will first introduce how computational neuroscience work, from a brief description of the brains and neurons to the way we model them, in order to understand the objects we will manipulate, and how they aim to represent some realities.
Then, we will present the tool we will use to analyze those networks : dynamical system theory. This field in general and more specifically the Lyapunov exponents and the dissipation will be of great help to unravel some of the specificities behind our networks.

After those introductions, we will start our actual research with a case aiming to represent to better simulate a macroscopic brain pathology : epilepsy. This part will be our link between the world the models aim to represent and the analysis of the models themselves, permitting us to see some important differences between different models of single neurons, and the influence of the connectivity in the networks. Most of all, this will show us the influence of randomness, and how we sometimes only master the networks we use in a statistical way.

Then, we will focus specifically on the study of models themselves, aiming to understand the difference between Driven and Self-sustained networks : similar in their activity except one requires an external drive to function properly while the other does not. This study will make us really use the dynamical systems tools we talked about previously, and show major differences between the two networks that would not be obvious without a thorough study.

The previous study is focused on the so-called Lyapunov exponents which, while interesting, are limited when it comes to studying very big systems such as the one we usually manipulate. This is why the next part is focused on the use of the Dissipation, which allows us to continue the previous research with a new approach, showing previously unheard similarities and differences.

In the end, in this thesis, we will not aim to "understand" or "conceptualize" neural networks as a whole. We will not directly aim to use them to understand reality either (although the first part of our results can create very good insights to understand epilepsy).
Our main goal will be to understand the limitations of our neural network model, and how apparently similar networks could actually differ. All models are wrong, some are useful. This work aims to delimit the frontier of usefulness a bit more than it was known before.

# 2 Computational neuroscience

## 2.1 How does the brain work ?

The human brain is a remarkable organ responsible for our thoughts, emotions, behaviors, and bodily functions. It consists of a huge amount of interconnected nerve cells called neurons and supportive glial cells ($10^{11}$ neurons and roughly the same number of glial cells in the human brain (von Bartheld, Bahney, & Herculano-Houzel, 2016)). Together, they form a complex network that allows for information processing and communication within the brain.

When looked globally, from the outside, we can observe global states : the brain exhibits different patterns of activity depending on the state of consciousness : here we will only developed awakeness and sleep.
While those different states are easy to qualify, the specifics were harder to quantify precisely. To do so, on top of invasive methods with obvious limits, non invasive methods for analysing the whole brain were developed, such as EEG (Michel & Brunet, 2019), MEG(Gross, 2019) or fMRI(Chow, Wu, Webb, Gluskin, & Yew, 2017).

During wakefulness, the brain is in a state of heightened activity characterized by high-frequency, low-amplitude electrical patterns known as beta waves (13-30$Hz$). This state is associated with alertness, focused attention, and active cognitive processing.
While awake but resting, without specific focus, the brains idles a bit and become characterize by a lower frequency pattern : alpha waves (8-13$Hz$).
In contrast, during sleep, the brain transitions through different stages with distinct characteristics : non-rapid eye movement (NREM) sleep, which consists of several stages, including light sleep (stage 1) and deeper sleep (stages 2 and 3), and rapid eye movement (REM) sleep. NREM sleep is characterized by slow-wave activity (delta waves, around 1-4$Hz$) in the brain's electrical patterns, reflecting a more synchronized and restorative state. REM sleep, often associated with dreaming, is characterized by desynchronized brain activity resembling wakefulness.

These different global states of wakefulness and sleep serve important functions for brain health, cognitive processes, memory consolidation, and overall well-being.

Zooming a bit inside of the brain : at a macroscopic level, the brain can be divided into several major areas, each with specific functions. One of the most prominent areas is the cerebral cortex, which is the outer layer of the brain. The cortex plays a crucial role in higher-order cognitive processes, such as perception, attention, memory, language, and decision-making. It is divided into four main lobes: the frontal lobe, parietal lobe, temporal lobe, and occipital lobe, each responsible for different functions.

Within these brain areas, we find cortical columns (Mountcastle, 1957). Cortical columns are vertical columns of neurons that span through multiple layers of the cortex. They are specialized for processing specific types of information, such as orientation, color, or motion. Cortical columns enable the brain to perform complex computations and integrate sensory inputs.

At an even smaller scale, individual neurons are the fundamental units of the brain that we will take some time to detail in the next part. Glial cells, such as astrocytes, oligodendrocytes, microglia, and ependymal cells, provide support and maintenance functions for neurons, contributing to their well-being and efficient functioning.

## 2.2 Description of single neurons

Neurons are one of the fundamental bricks that construct the brain, and the core of our analysis. They were first observed and characterized by Santiago Ramón y Cajal in the beginning of the 20th century. A wide variety of neurons exist, and we will not describe them here, only pointing out that important differences can arise depending on what we want to describe. An example of a neuron, a pyramidal cell, is given at Fig.1.A). They are electrophysiological bodies that allow communication and transmission of information in the brain.
We will just present the basics of neuron anatomy, as in Fig.1B). The main parts of neurons are :

- Cell Body (Soma): The cell body is the main part of the neuron that contains the nucleus and other cellular organelles. It integrates incoming signals from dendrites and generates outgoing signals along the axon.

- Dendrites: Dendrites are branching extensions that receive signals from other neurons or sensory receptors. They increase the surface area available for receiving synaptic inputs and play a crucial role in transmitting information toward the cell body.
- Axon: The axon is a long extension of the neuron that carries the action potentials away from the cell body. At the end of the axon, there are specialized structures called axon terminals, which form synapses with other neurons to transmit signals.
- Synapses: Synapses are junctions between neurons where information is transmitted from one neuron to another. They can be either chemical or electrical. In chemical synapses, neurotransmitters are released from the presynaptic neuron's axon terminals, cross the synaptic gap, and bind to receptors on the postsynaptic neuron's dendrites or cell body. Electrical synapses allow direct electrical communication between neurons through gap junctions.
- The myelin sheath and the nodes of Ranvier: Nodes of Ranvier are small gaps in the myelin sheath - a fatty substance made by some glial cells and responsible for increasing the speed of the electrical signal - along the length of the axon. They facilitate the propagation of action potentials by allowing the electrical signal to jump from one node to another, a process called saltatory conduction.

The membrane is present in all cells, including neurons. It is the specific part that will be responsible to exchange ions with the exterior of the cell, allowing for variation in the polarization of the membrane which, as we will see, allows for the creation and transmission of the action potential.
From now on, we will mainly focus on the representation of the membrane. For more detail on the neuron anatomy, one can refer to (Martin, 2012).

The membrane of neurons contains various ion gates, also known as ion channels, which are specialized proteins that regulate the flow of ions across the neuron membrane. These ion channels play a crucial role in generating and propagating the action potential : the electrical signal responsible for the transmission of information in the brain. There are various channels in the membrane, the principal beings :

- Sodium channels: Sodium channels allow the passage of sodium ions (Na+) into the neuron. They play a key role in the initiation and propagation of action potentials. Sodium channels have different states, including closed, open, and inactivated states.
- Potassium channels: Potassium channels facilitate the movement of potassium ions (K+) out of the neuron. They are involved in repolarizing the cell membrane after an action potential, restoring the resting membrane potential. Potassium channels contribute to maintaining the balance of electrical charge and play a role in controlling the excitability of neurons.
- Calcium channels: Calcium channels allow the entry of calcium ions (Ca2+) into the neuron. They are involved in various cellular processes, including neurotransmitter release, synaptic plasticity, and gene expression. Calcium influx through these channels is essential for triggering cellular responses and regulating neuronal activity.

These ion channels are dynamic and can be regulated by various factors, including voltage changes, neurotransmitters, and intracellular signaling molecules. The opening and closing of ion channels control the flow of ions, which, in turn, determine the electrical properties and signaling capabilities of neurons. While it is crucial for the functioning of the action potential, we will not talk again of Calcium channels, only focusing on the Sodium and Potassium ones as they are directly responsible for the formation of the action potential itself, and not its consequences.

Focusing on the action potential : when sodium gates open, sodium ions can enter the neuron, causing a depolarization which in turn opens more voltage sensitive channels, causing an acceleration of the depolarizing strong enough it can propagates to the rest of the axon. Then, the potassium gates open, hyperpolarizing the neuron which goes back to a resting state. The action potential propagates through the axon, and then allows the release of a quantum of chemical bodies outside of the neuron, through the synapses connecting the action of the neuron to the dendrites of other neurons. Those chemical bodies will then modify the membrane potential of the new neuron, depolarizing or hyperpolarizing it, influencing its probability to spike. For more details on the mechanism of spike and the general biophysiological presentation of the neuron, look at (Johnston D, 1994).
The goal of this brief and simplified description is to give the reader the keys to understand the model we use later.

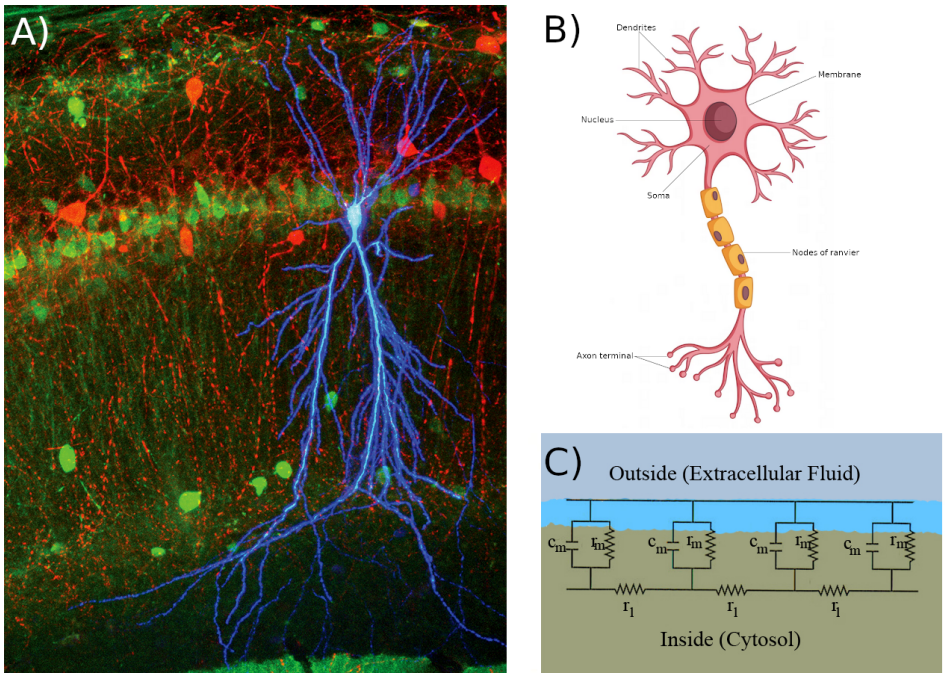


Figure 1: A) Picture of a neuron (a stained pyramidal cell in mouse hippocampus), Courtesy: National Science Foundation. B) Schematic, simplified representation of neurons (from www.freepik.com). C) Electric representation of the membrane of neurons (that will be simulated by corresponding differential equations), from Michael Klausen, Public domain, via Wikimedia Commons.

## 2.3 Models of single neurons

Half a century after the first observation of neurons, the famous paper of Hodgkin and Huxley (Hodgkin & Huxley, 1952a) came, proposing a mathematical model of the squid giant axon which would reproduce its action potential. This model, as many other, simulate the membrane of the neurons, modelling it as an RC circuit as in Fig.1.C) with some added complexity.
From that time, a lot of different models of single neurons were proposed, for various uses. The reader can see a great introduction to computational models in neuroscience in (Sterratt, Graham, Gillies, & Willshaw, 2011).
Those neuronal models add a lot of difference, and we will focus on two of them.

### 2.3.1 Major differences and classification between models of single neurons

First, models could be more or less "realistic", or to be precise "biologically realistic". This is already a complicated term because, as we said previously, models never reproduce reality perfectly. Here, it actually means the model is a mechanistic one, that we try to understand how the neuron actually works in real life, and to represent each important part in a simplified way, combining the whole in order to obtain an emergent activity similar to what we observed. This would serve two purposes : first, it would validate the other, smaller models and representations that were used to create this emergent mechanistic model. Then, it would allow direct correspondence to test specific hypotheses, and it would allow a clearer understanding of any found phenomenon. A prime example is the Hodgkin Huxley model we presented before. Here, the different gates and their probability of opening are modelled, allowing a complex and yet easy to grasp dynamic to arise and to model similar to experiment action potential. Changing one parameter would be easy to interpret and possible to test, allowing a quick back and forth between experiment on models and on live tissues.
On the other hand, models could be phenomenological, focusing on reproducing a specific phenomenon, but not on understanding why it happens or how exactly the different terms of the model translate to real life. The end of that spectrum could be the neural networks used in machine learning, that no longer tries to connect the parameters of the equations used to represent the neuron activity to real life data, but only aim to optimize specific functions as well as possible. We could also mention Izhikevich's quadratic model (E. M. Izhikevich, 2007) that only serves to reproduce the shape of the action potential without caring about the structure of the neuron. To some extent, this is also the case of integrate and

fire models, at least compared to the Hodgkin and Huxley model. As they do not need to model as many things as more realistic neurons, the equations that represent those models are often easier to use and less computationally costing, allowing for long simulations, big networks (as we will see later), or powerful mathematical tools to analyze them as in (E. M. Izhikevich, 2007). Most models would be somewhere on the spectrum, but it is useful to remind the reader that they can both be useful, but generally not for the same purpose.

Second, apart from the "realism" of the model, and among other important differences, we want to point out that neuronal models do not reproduce every and all behavior of neurons. They generally allow to reproduce some specific range of behaviors, but cannot represent others. A good example of that is the AdEx model (Brette & Gerstner, 2005a) that we will define later that can, thanks to an added part modelling the "adaptation", show a huge range of behavior, specifically bursts (Naud, Marcille, Clopath, & Gerstner, 2008a) that are not normally accessible for other neuron models, such as Hodgkin Huxley. We can have models that reproduce very well some specific behavior, but fail to reproduce others, and others that do not reproduce them as well but have access to a lot more behaviors. We could also have models that can do both, of course, but they are generally more complex and computationally costly, so a choice needs to be made.

### 2.3.2 What do the models represent ?

The computational model we will use in this work will represent action potentials with different sets of equations, but both of them will use differential equations showing the evolution of the membrane potential.

AdEx model is part of the integrate and fire types of models, so it does require an "artificial" reset after a certain threshold of that membrane potential Vm is past, while Hodgkin and Huxley model do not need it.
Still, for both of them, it is when the Vm goes through a threshold that the spike is counted and has effect on the network, which makes sense considering this is what happens in real neurons, although we do not represent the propagation of the signal in the axon. But as we said before, no representation is perfect, and they all serve a general purpose within certain limits.

While both of those equations are supposed to represent Vm, they use different details to do so that interact in specific ways with the rest of the equations. Therefore, can we actually say that the Vm from the Adex and the one from another model, such as Hodgkin Huxley, are the same ?
It seems obvious that they are not, but does that mean we can't compare them ? Probably not either, as they are supposed to represent the same object, just in a different way. It is outside the scope of this work to go deep in the ontology of the objects we model, but it is interesting to note that there is generally no question in comparing apparently similar objects without questioning if such comparison is actually possible in the first place.
While we will indeed compare the different models and networks, this idea of the potential differences between apparently similar objects is the core of that research, and it helps to think it is a more general problem than we think at first, as it could give us more epistemological carefulness.

### 2.3.3 In practice : how are the models defined ?

Now that we are over with the theoretical and philosophical part, we will focus on the practical models we will use. Those different models are so-called point neurons models, meaning we do not consider the morphology of the neurons, the axon, the soma and the dendrites, and that our representation will be a 1D one instead of a 3D object. It is useful to see those models as this : there is an input, the input is transformed by the equations of the model, and then it produces an output. There is no spatiality, only time is meaningful here. Let's see the details of the models !

**Type of population** When we create a network, all neurons follow the same model and therefore the same equations, as the ones we will present later. On the other hand, neurons are not all identical in the brain. While it is out of the scope of this study to try to represent all the types of neurons in the brain (for such work, see (Markram et al., 2015) models of some simple types of neurons exist (E. M. Izhikevich, 2007). The most basic and broad categories, and the most useful to have a wide variety of dynamics, are excitatory and inhibitory neurons.

As all neurons, once they have an action potential, they modify the membrane potential of some other neurons they are linked to. The difference lies in the modification : excitatory neurons will make the post-synaptic neurons more likely to fire by increasing their membrane potential, exciting them, while inhibitory neurons will decrease their membrane potential and therefore the probability to fire, inhibiting them.

Then, it is a matter of balance. If the network is too inhibited, whether because there are too many inhibitory neurons or because their inhibition is too strong, then the network will just inhibit itself and stop all activity. On the other hand, if the excitation is too strong, the network will "explode" and will be always active. Due to the nature of our simulation, it means it will alternate between having an action potential, going back to a reset value, wait for a refractory and fire again, leading to an average firing rate of the inverse of the refractory period (which, for our case, would be 200Hz).

There are different kindss of excitatory and inhibitory neurons, but for simplification we use one of each. All our excitatory neurons are so-called Regular Spiking (RS) neurons: they have a fairly low firing rate. On the other hand, the inhibitory neurons are so-called Fast Spiking (FS) neurons and, as the name suggests, have a (much) high firing rate.

**Spiking neurons model : AdEx** The Adaptive Exponential integrate and fire model, or AdEx, is, as we previously said and from its name, a type of integrate and fire. This means we do not model the whole spike, but only the time upward part before arriving at a threshold, and then the membrane potential Vm is artificially put back and clamped at a lower value for a few (5) milliseconds. Then, a spike is generated and has some effects that we will describe later.
Each neuron in the AdEx network is described by Eq.(39) and Eq.(42) as follow :

$$C_m \frac{dV_m}{dt} = g_L(E_L - V_m) + g_L \Delta_T \exp\left(\frac{V_m - V_T}{\Delta_T}\right) - w + I_{syn} \tag{1}$$

$$\tau_w \frac{dw}{dt} = a(V_m - E_L) - w$$

When the membrane potential crosses a threshold, a spike is emitted, and the system is reset:

$$\text{if } V_m \geq V_D \quad \text{then } \begin{cases} V_m \to V_R \\ w \to w + b \end{cases} \tag{2}$$

The first differential equation describes the evolution of the membrane potential. To begin with, we have the leaky integrate and fire part of AdEx, with $C_m$ the membrane capacitance (often equal to $200pF$), $Vm$ the membrane potential (in $mV$, $g_l$ the leak conductance (around $10nS$), $E_L$ the leak reversal potential (around $-65mV$, all being analog to RC circuits. That part is an attractor to the value $E_L$, lower than the spike detection threshold. This means that without external current, the neuron would never spike. This is the role of the additive synaptic current $I_{syn}$ that correspond to the network activity and that could increase or decrease the membrane potential depending on the excitatory/inhibitory balance.
Then, we have the "exponential" part of AdEx, that specifically aims to represent the upward part of the spike better when the membrane potential is high enough and the neuron is on the way to spike. This part will introduce a new "exponential" threshold $V_T$ (around $-50mV$), different from the spike threshold. The latter will be responsible for the spike detection and is explained in Eq.(42), while the former is an additive term that represents the sudden acceleration of the depolarisation that will lead to a spike after a certain value. On top of $V_T$, we introduce $\Delta T$ the spike sharpness ($2mV$ for excitatory neurons, and $0.5mV$ for inhibitory ones).
Finally, we have the adaptation part of AdEx : $w$ (in $pA$). $w$ being a positive value, it will tend to hyperpolarize the membrane potential after a spike, meaning it will be harder for a neuron to spike a second time immediately after a first spike : this is adaptation. As the second part of Eq.(39) shows, $w$ will leak to smaller values, meaning the diminution in membrane potential is just temporary. It will leak more or less quickly depending on $\tau_w$, the adaptation time constant (between $100ms$ and 1000ms), and it will also depend on $a$, the adaptation conductance (often put to 0 in this work). Finally, as shown in Eq.(42), $w$ increases by a value $b$, the adaptation current increment (very variable, between 0 and $100pA$) when there is a spike.

As said previously, Eq.(42) shows the consequences of a spike for the neuron : we introduce the spike detection threshold $V_D$ that has various value, often between $-45mV$ and $-47.5mV$, but some models use as high as $+20mV$ to see the complete spike. Of course such variation would change the dynamic of the network as it would increase or decrease the frequency of spikes. We already talked about the effect on the adaptation $w$, what is left is the resting potential $V_R$ (often at $-65mV$) at which the spike is clamped right after a spike for a refractory time $T_{ref}$ (not shown in those equation, $T_{ref} = 5ms$).

Finally, we will also use the CAdEx model in this thesis, a derivative of the AdEx model. More precision will be given in the appropriate section, as we want to emphasize the difference between AdEx and Hodgkin-Huxley models here.

**Spiking neurons model : HH** From here, we will call the Hodgkin-Huxley model HH.
Contrary to before, HH is not an integrate and fire model, and the whole action potential is represented. While not necessary (as there exist models that directly propagate the action potential), it will also registered a spike after an artificial threshold, and the network will receive this output in a similar way as for AdEx networks.
The HH model follows :

$$C_m \frac{dV_m}{dt} = g_L(E_L - V_m) + g_K n^4 (E_K - V_m) + g_{Na} m^3 h (E_{Na} - V_m) + I_{syn} \tag{3}$$

with gating variables (in ms):

$$\begin{aligned}
\frac{dn}{dt} &= \frac{0.032(15. - V_m + V_T)}{(exp(\frac{15. - V_m + V_T}{5.}) - 1.)}(1. - n) - 0.5exp(\frac{10. - V_m + V_T}{40.})n \\
\frac{dh}{dt} &= 0.128exp(\frac{17. - V_m + V_T}{18.})(1. - h) - \frac{4.}{1 + exp(\frac{40. - V_m + V_T}{5.})}h \\
\frac{dm}{dt} &= \frac{0.32(13. - V_m + V_T)}{(exp(\frac{13. - V_m + V_T}{4.}) - 1.)}(1 - m) - \frac{0.28(V_m - V_T - 40.)}{(exp(\frac{V_m - V_T - 40.}{5.}) - 1.)}m
\end{aligned} \tag{4}$$

To sum it up, that gates variables dynamics are sometimes written as

$$\begin{aligned}
\frac{dn}{dt} &= \alpha_n(V(t))(1 - n) - \beta_n(V(t))(n) \\
\frac{dh}{dt} &= \alpha_h(V(t))(1 - n) - \beta_h(V(t))(h) \\
\frac{dm}{dt} &= \alpha_m(V(t))(1 - m) - \beta_m(V(t))(m)
\end{aligned} \tag{5}$$

The first equation describes, as previously, the evolution of the membrane potential. There are identical terms that have the same range of values. The specificity of HH, of course, is that it takes into account various gates that allow for a depolarization or hyperpolarization of the membrane. Those are the Potassium current and the Sodium current.
The first one introduces the terms $g_K$ and $E_K$, the potassium conductance and reversal potential (typically $6nS$ and $-90mV$). Due to the low value of $E_K$, this current will hyperpolarize the membrane potential. The strength of that hyperpolarization will depend on the term $n^4$ a dimensionless probability of activation the potassium channels, and whose evolution is described in Eq.(44).
The second one introduces the terms $g_{Na}$ and $E_{Na}$, identical as before but for sodium (typically $20nS$ and $+60mV$). Contrary to before, the sodium current depolarizes the membrane potential, and the modelization of its gate is a bit more complex, as it depends on two different terms : $m$ and $h$ (also dimensionless probabilities). The first one is similar to $n$ and corresponds to the activation of sodium channels, but the second one is the opposite and corresponds to the inactivation of those channels.

While we will note go further into the evolutions of $n, m, h$ we note that they are depending, again, on the difference between a reversal potential $V_T$ ($-50mV$ for excitatory neuron, $-52mV$ for inhibitory ones) and the membrane potential $V_m$.

For this model, there is no external reset, but we still have a detection spike threshold $V_D$ as for AdEx, although it has no effect on the intrinsic dynamic of the neuron.

**Differences and similarities** Both models only model the dynamic of the membrane potential. When it is high enough, the simulation considers a spike is done and its consequences start, but are not properly modeled : there is no model of the quanta of chemicals that will be delivered from the dendrites to the axons of other neurons, only an increment in some conductance. They are also point neurons models, which means we do not consider them to have spatial dimensions : all the dynamic happens in a point and we do not consider travel time, including to deliver the «spike information» to other neurons. Please note that all values of parameters are within biophysical range (see (Hodgkin & Huxley, 1952a; Hille, 1992; Naud et al., 2008a; Górski, Depannemaecker, & Destexhe, 2021)). A major difference between those two models is that Adex is part of the integrate and fire family of models, meaning there is a model of the beginning of the action potential, and then a reset and a rest that is not modeled. HH on the other hand reproduce the whole action potential, including the downward phase.
As we can see, another major difference is that Adex is a 2 dimensional model, while HH is a 4 dimensional one.
There is also a conceptual difference that synthesize the last two points : HH was made as a model to reproduce biological reality as well as possible. It therefore models what we know (or knew) of neurons : the gates activation and deactivation that will result in the potassium and sodium current which have antagonist effects and create the shape of our action potential. Adex on the other hand aim to reproduce the phenomenon, caring less about how its components can be translated to the «reality» in a bijective way (although there are still possibilities to do so and there exist way more phenomenological models). Due to its simplified nature, Adex is on the other hand able to represent another behavior that does not exist with HH : adaptation. It allows to create burst of activity, or varying activity with a time interval between spikes that is not constant (growing bigger or smaller depending on the parameters use) which allow for a wider variety of phenomenons (for more details on what behaviors Adex can reproduce, see (Naud et al., 2008a))

## 2.4 From neurons to networks

As explained before, neurons have synapses that are used to interact between each other. While we consider point neurons and therefore no proper synapses, we still make a link between the neurons.
We start by creating a graph with all the links between the neurons. Links are unidirectional, so if neuron A is linked to neuron B, neuron B is not necessarily linked to neuron A. We use conductance based model, which means that the effect of said link is that when neuron A spikes, neuron B will receive instantaneously (actually, in the next time step) an increase in a specific conductance, which will then modify its current. We use different nernst equilibrium to translate the excitatory or inhibitory nature of the link, as follow :

$$I_{syn} = g_E(E_E - V) + g_I(E_I - V) \tag{6}$$

With $E_E$, $E_I$ the excitatory and inhibitory reversal potential (often at $0mV$ and $-80mV$), and $g_E$, $g_I$ the excitatory and inhibitory conductance. As said previously, those conductances are the ones that receive an increment when a connected neuron spikes. If it is an excitatory neuron, $g_E$ will increase. As the threshold is below $E_E$, the potential difference in the parenthesis will always be positive, and therefore this contribution will increase the membrane potential value. On the other hand, if there is a spike from an inhibitory neuron, as $E_I$ is always below the resting potential, making the difference of potential always negative, an increase in $g_I$ will lead to a reduction in the membrane potential.
On top of that, those conductances are modeled as leaky currents, so we have two equations showing how they evolve :

$$\frac{dg_{E/I}}{dt} = -\frac{g_{E/I}}{T_{E/I}} \tag{7}$$

With $T_{E/I} = 5ms$ the excitatory/inhibitory decay.

There are various ways to organize the graph between neurons. First, we could think of the value of the link. We could have a variation on the value of the link, or on a matrix, which would show how strongly or weakly each neuron could be connected. We choose a simple « all or nothing » case where neurons have a connectivity of either 0 or 1, and nothing in between.
On top of the value, we can also think of how to arrange the links. For example, we could try to reproduce some specific patterns of activity and implement an adaptation rule, such as STDP (Hebb, 1949)for the neuron connection to change. We could also have some influence on the topology, such as a simple « neurons that are farther from each other are less likely to be linked », which leads to specific results (such as wave activity) (Davis et al., 2020). Here, we chose a simpler form : an Erdős–Rényi graph, meaning all

connections are random, independent from any notion of topology. To be precise, we have a 5% chance that any two neurons are connected. This leads to an apparent lack of structure, but it is interesting to see that there will still be differences between neurons as can be seen in Fig.2, some of them having more or less links than the average, which could strongly influence their dynamics.

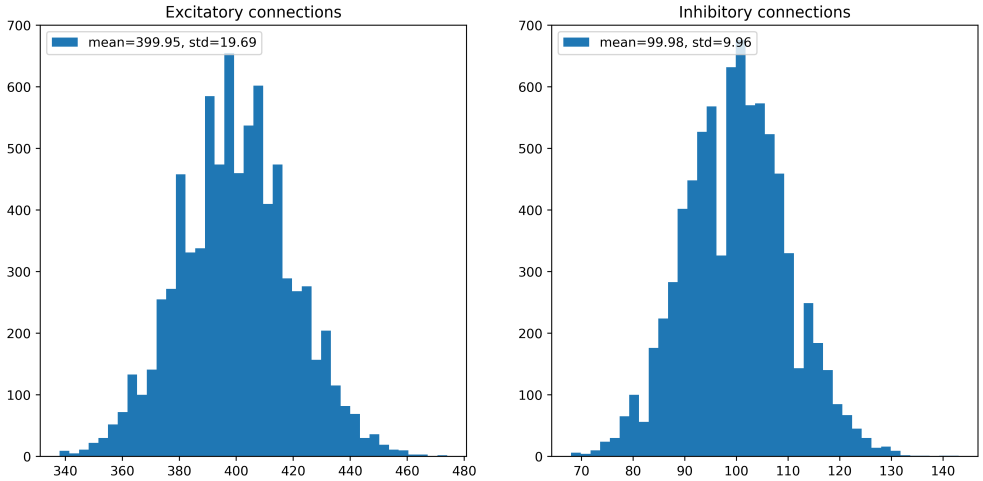


Figure 2: Example of the distribution of the excitatory (left) and inhibitory (right) connections for each neuron in the network we use. There is typically 400 excitatory connections per neuron, with an std of 20, and 100 inhibitory connections with an std of 10.

It also matters to choose how many neurons we want to consider, knowing the goal is not to represent a specific object (like a cortical column as in (Markram et al., 2015)) but simply something on the range of a few thousands to tens of thousands of neurons, to have relevant behaviors that only arise when the scale is big enough. Here, we consider populations of 10 000 neurons, 80% being excitatory neurons and 20% being inhibitory, within range of what is observed experimentally (Wonders & Anderson, 2006; Alreja, Nemenman, & Rozell, 2022).

## 2.5 Emergence, chaos and complex dynamics : behavior of networks

The global idea of the link between single neurons and networks, what it produces and why it could be complex, can be sum up with a cartoon made in Fig.3

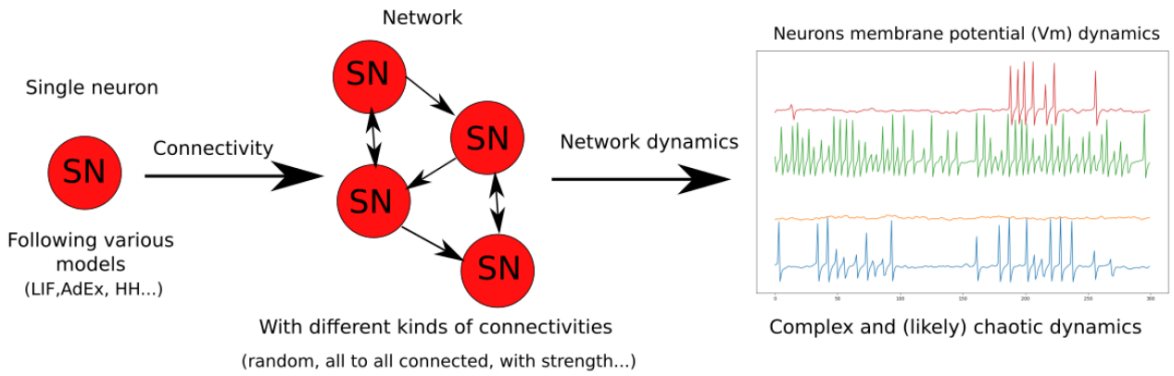


Figure 3: Cartoon of the path from single neuron to network activity

The idea is that the resulting interaction of the neurons in the network leads to an enormous space of complex emergent dynamics. To help understand it, and while looking for said emergent dynamic in real brains would be very relevant, we decided to focus on analysing simulations of a couple of models. Our goal is to develop tools to better understand the behavior of the network at different scales, despite the

issue of it evolving in a very high dimension such as the so-called curse of dimensionality (Köppen, 2000). There are some obvious candidates to do that, such as studying the membrane potential of all neurons. There are actually two issues with this approach : the first one lies in the enormous phase space to consider, which gets even worse when we consider networks of thousands or tens of thousands of neurons. The second issue is the one we developed earlier : as it is represented in different ways within different models, can we actually say we are studying the same object ?
We could also binarize the phenomenon and only consider whether there is a spike or not at each time window we want to consider. While it simplifies drastically the representation of the activity, it also raises questions about how much is lost by said simplification. If we consider all the neurons in a big network, it also stays way too big to analyse, losing potentially precious information for a still very hard analysis. Then, we can think of using averages of quantities, such as the firing rate of the network, which is indeed was easier to analyse. While an interesting approach, it obviously lose a lot of information, and mainly makes it difficult to know if we can still capture important emergent phenomena with it. If we want to average things, we need a tool that is made for it and that depends on the activity itself, and not on part of the network we are trying to modelled. To do that, we take the approach of looking at neural networks as dynamical systems.

**The question of information** As we have said, the main interest of neural networks is the spike production of individual neurons, and how those spikes will influence the network in the next place. There is a constant exchange of scales between the single neurons and the whole network, always influencing each other. We will call that spike production the "activity" of the network, and we will often link it to the Firing Rate (FR) of the whole network, showing an average of this activity. Of course, this will mainly inform us of the quantity of spikes and its variation, assuming this is where the meaningful information produced by the network is. But there are actually lots of paper following a different hypothesis, claming information is actually in the so called "neural coding" (Borst & Theunissen, 1999; Azarfar, Calcini, Huang, Zeldenrust, & Celikel, 2018), a complex term that englobes a lot of ideas including the one claiming that the precise timing of the single neuron spike is what matters more than a quantity. It is not the goal of this thesis to go further with this hypothesis : while it could be that the spike timing is important to transmit specific information, the networks we use are not trained to reproduce any specific kind of output apart from a similar FR as the ones observed in real data. This is also what will we do in this work, focusing on the activity as the amount of spike produce, and observing what is linked to the change of that activity.

**Different kind of networks behaviors** Now that we have talk about what we meant by activity and why it was important, what kind of activity can the network produce ?
In (Brunel, 2000b), Brunel introduced a typology of the dynamics of brain activity patterns. It involved :

- Asynchronous Irregular: In this pattern, individual neurons in a network exhibit irregular firing activity with no synchronized firing among neurons. The firing times of neurons are highly variable, and there is no clear pattern of coordination or synchronization. This activity pattern is often associated with a balanced network state where excitation and inhibition are finely tuned.
- Synchronous Irregular: In this pattern, neurons in a network fire in synchrony, but the firing activity is still irregular. The firing times of neurons may exhibit some level of synchronization, but the overall activity remains irregular. This pattern can arise from network interactions and can occur, for example, during specific cognitive tasks or in certain pathological conditions.
- Asynchronous Regular: In this pattern, individual neurons exhibit regular firing activity, but there is no synchronization or coordinated firing among neurons. Each neuron fires at a consistent rate, but there is no temporal correlation between the firing of different neurons. This pattern may arise in specific neural circuits or under certain experimental conditions.
- Synchronous Regular: In this pattern, neurons in a network fire in synchrony, and their firing activity is regular and highly correlated. The firing times of neurons are precisely coordinated, often in rhythmic patterns. This pattern can be observed in various brain states, such as during certain oscillatory activities like gamma or theta rhythms, or during specific cognitive processes like attention or perception.

An illustration of some of those states is given is Fig.4

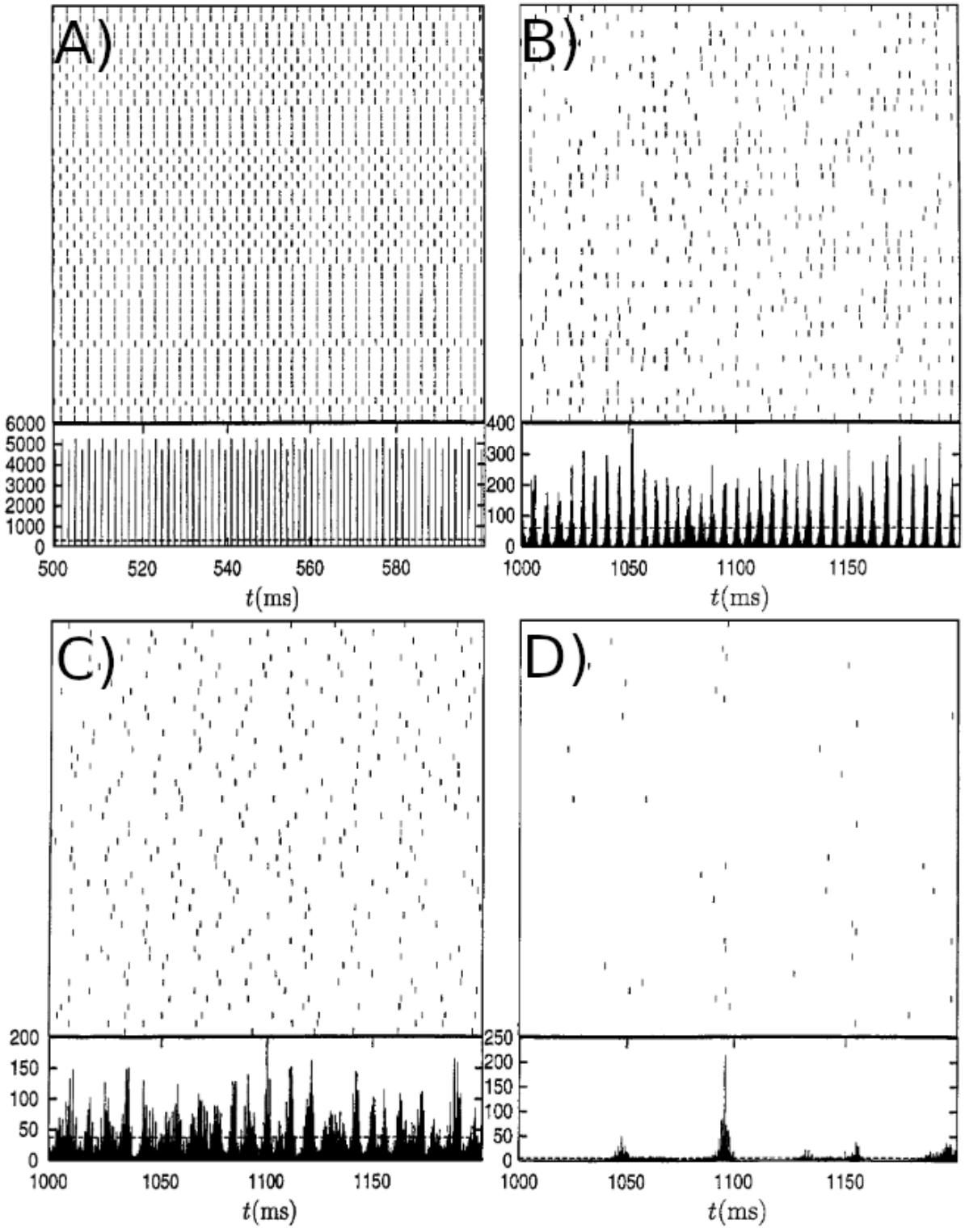


Figure 4: Illustration of Brunel typology, with raster plots (up) and firing rates (down).
A) : Synchronous regular. B) : Fast oscillation, Synchronous Irregular. C) Asynchronous Irregular. D) Slow oscillation, Synchronous Irregular

It's important to note that these patterns are simplifications and abstractions used to describe the dynamics of neural networks. The actual brain activity is highly complex and can involve a combination of these patterns across different brain regions and under various physiological or cognitive states.

We will focus here on AI states, as that type of activity is similar to that observed in awake animals

(Matsumura, Cope, & Fetz, 1988; Steriade, Timofeev, & Grenier, 2001; Destexhe, Rudolph, & Paré, 2003; Lee, Manns, Sakmann, & Brecht, 2006).

## 2.6 Taking a step back : the Mean-field

**What is a mean-field ?** Mean-field, and specifically those of neural networks, are mathematical frameworks used to understand the collective behavior of large populations of neurons. These tools provide a simplified representation of the complex dynamics occurring within neural networks, focusing on the average or mean activity of the population rather than individual neuron behavior, and allow to use thorough analysis impossible to use if each single neurons were considered. While at first neural network mean-field were studied such that the neurons are from a unique population and are (asymptomatically) independant (Sompolinsky, Crisanti, & Sommers, 1988), many have since then used different, correlated populations and developed strong physical analysis of said neural network from the mean-field (Faugeras & MacLaurin, 2015; van Meegen, Kühn, & Helias, 2021) and even created toolbox to use the mean-field analysis of neural network easily (Layer et al., 2022).
By approximating the interactions between neurons with statistical measures, such as firing rates or membrane potentials, mean-field models can also describe the network's global dynamics using differential equations or rate equations, which can be used as a simplified version of the networks dynamics. This mean-field approach enables us to study phenomena such as synchronization, oscillations, and transitions between different network states (Muller, Reynaud, Chavane, & Destexhe, 2014; Capone, Volo, Romagnoni, Mattia, & Destexhe, 2019; Goldman et al., 2023). It has been applied in various research areas, including sensory processing, motor control, memory, and cognition. Mean-field models capture the emergence of collective behaviors, such as pattern formation and decision-making processes. They offer a computationally tractable framework for investigating large-scale neural dynamics and making predictions about network behavior under different conditions.

It is important to note that mean-field models simplify the intricate details of individual neuron behavior and local interactions. Nevertheless, they provide valuable insights into the global behavior of neural networks, helping us understand how large populations of neurons process and transmit information.

**Mean-field model of AdEx networks** We used a mean-field models of AdEx networks, using the model defined in (Volo, Romagnoni, Capone, & Destexhe, 2019). This mean-field model was originally based on a Master Equation formalism developed for balanced networks of integrate-and-fire neurons (El-Boustani & Destexhe, 2009). This model was first adapted to AdEx networks of RS and FS neurons (Zerlaut, Chemla, Chavane, & Destexhe, 2017), and later modified to include adaptation (Volo et al., 2019). This latter version corresponds to the following equations (using Einstein's index summation convention where sum signs are omitted and repeated indices are summed over):

$$T\frac{\partial \nu_\mu}{\partial t} = (F_\mu - \nu_\mu) + \frac{1}{2}c_{\lambda\eta}\frac{\partial^2 F_\mu}{\partial \nu_\lambda \partial \nu_\eta} \tag{8}$$

$$T\frac{\partial c_{\lambda\eta}}{\partial t} = \delta_{\lambda\eta}\frac{F_\lambda(1/T - F_\eta)}{N_\lambda} + (F_\lambda - \nu_\lambda)(F_\eta - \nu_\eta) + \frac{\partial F_\lambda}{\partial \nu_\mu}c_{\eta\mu} + \frac{\partial F_\eta}{\partial \nu_\mu}c_{\lambda\mu} - 2c_{\lambda\eta} \tag{9}$$

$$\frac{\partial W}{\partial t} = -\frac{W}{u_w} + b\nu_e + a(\mu_V(\nu_e, \nu_i, W) - E_L) \tag{10}$$

Where $\mu = \{e, i\}$ is the population index (excitatory or inhibitory),$\nu_\mu$ the population firing rate and $c_{\lambda\eta}$ the covariance between populations $\lambda$ and $\eta$. $W$ is a population adaptation variable (Volo et al., 2019). The function $F_{\mu=\{e,i\}} = F_{\mu=\{e,i\}}(\nu_e, \nu_i, W)$ is the transfer function which describes the firing rate of population $\mu$ as a function of excitatory and inhibitory inputs (with rates $\nu_e$ and $\nu_i$) and adaptation level $W$. These functions were estimated previously for RS and FS cells and in the presence of adaptation (Volo et al., 2019).

At the first order, i.e. neglecting the dynamics of the covariance terms $c_{\lambda\eta}$, this model reduces to:

$$T\frac{d\nu_\mu}{dt} \quad = \quad (F_\mu - \nu_\mu)\ , \tag{11}$$

together with Eq.(10). This system is equivalent to the well-known Wilson-Cowan model (Wilson & Cowan, 1972), with the specificity that the functions $F$ need to be obtained according to the specific

single neuron model under consideration. These functions were obtained previously for AdEx models of RS and FS cells (Zerlaut et al., 2017; Volo et al., 2019) and the same are used here.

## 2.7 Autonomy in networks : Driven vs self-sustained

On top of those two models, we also represented what we call here two types of networks : driven and self-sustained networks, and that will be more detailed in chapter 3.
Driven networks are the ones that are usually used : the network requires an external (here poissonian) input to produce an activity (see Fig.5(a-b). The latter on the other hand only needs an initial "kick" to start, and will, as the name suggests, have a self-sustained activity after (see (see Fig.5(c-d)). Both types of networks are interesting for different reasons : driven networks might be more realistic, as a small network is never cut out of the rest of the brain, and has no specific reason to sustain itself. On the other hand, self-sustained networks allow for an easier analysis of their dynamics, as there is no external input and most of all no sources or randomness (through the poissonian input) that will affect them. Please not that, while some change of parameters and study of the dynamics of the system gives good insight on how to obtain a self-sustained network, it is beyond the scope of this thesis to prove they were actually self-sustained forever and it was only checked that they would have a seemingly stable activity for longer than the typical length of simulation we needed. While the self-sustained networks do not require it, we sometimes add a drive to them for the sake of comparison.

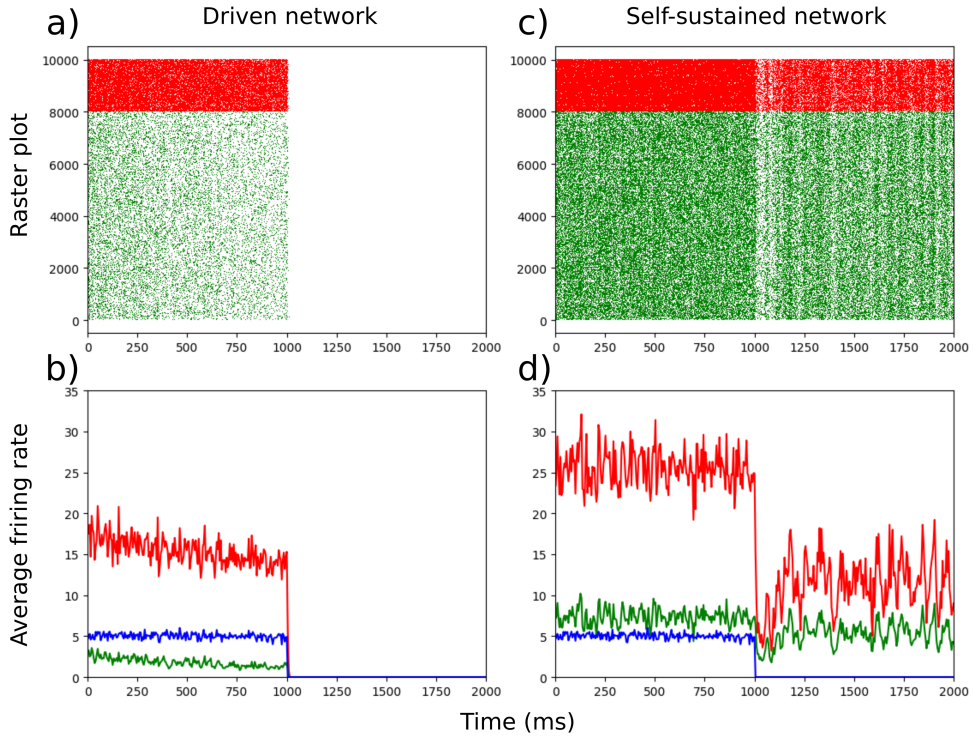


Figure 5: Illustration of driven networks (a) and b)) and self sustained networks (c) and d)) with an external poisson drive that stops at 1000ms. a) and c) represent raster plots, with green being RS neurons and red being FS neurons. b) and d) represent the average firing rate, with the same colour code and blue being the poisson input. As can be seen, when said poisson input stops, the driven networks immediately stop any activity, while the self-sustained reduce but maintain its activity.

## 2.8 Perturbations in neural networks

Finally, we need to introduce another concept. As we said previously, Driven networks require an external drive to have an activity. But in principle, we can always put an external input to a network. This input can be of various shape and form, and for various times. If it is always present, it will change the dynamic of the network, shaping a different attractor and a new stable state. This is the case for our driven networks, for example, but it would also be the case for self-sustained networks that use a drive. Studying those new states could be of great interest, showing how the networks react with different

amounts of constant drives, and could also lead to establish differences between networks, the goal of this thesis. But a constant noise is just a new stable state, and there is another kind of state that is interesting : transient ones. This arises due to perturbation, a shorter input that will temporarilly change the networks without caring if it stabilizes or not before stopping. Those perturbations can vary a lot, being more or less steep, more or less strong, more or less long, repetitive or unique, uniform or not, focusing on a group of neurons or to the whole network, etc etc. Each of those differences could produce various responses and represent various activity from the brain, whether healthy or pathological.
In this thesis, we will use perturbation following this form :

$$\begin{aligned} \nu_{pert}(t) = \beta + \alpha * (exp(-(t-T_1)^2/(2.*\tau^2)) * H(-(t-T_1)) \\ +H(-(t-T_2)) * H(t-T_1) + exp(-(t-T_2)^2/(2.*\tau^2)) * H(t-T_2)) \end{aligned} \tag{12}$$

where $H$ being the Heaviside function, $\beta$ the (potential) external drive (in the range of 0 to 5 $Hz$). This function takes the general form of a high plateau, where $T_1$ and $T_2$ are the times when the perturbation reaches its beginning and end respectively, $\alpha$ defines its maximal height and $\tau$ is the time constants associated with the exponential rise and decay of the perturbation.

# 3 Dynamical systems

In order to study our neuroscience system, we need a prism of observation. As the models will be mathematical ones, made of differential equations, and some models can exist without any added noise, we considered them as deterministic system instead of statistical ones (although that approach also does make a lot of sense, eg with the approach of entropy in the brain (Cofre & Cessac, 2014; Herzog et al., 2023)), which is why we naturally turn to dynamical system theory.

## 3.1 What are Dynamical systems ?

**In general**
Dynamical systems are mathematical models used to describe how certain quantities or variables change over time. These systems can represent a wide range of phenomena, from the motion of celestial bodies to the behavior of biological networks and the dynamics of financial markets. Dynamical systems are typically described by differential equations (or iterative equations), and their behavior can be analyzed using various mathematical tools. We will describe some of them in the next parts.

The general form of continuous-time dynamical systems is described by ordinary differential equations (ODEs), such as:

$$\frac{dX}{dt} = F(X, t) \tag{13}$$

Where $X$ represents the state variables of the system (either a single point or a vector), $t$ is time, and $F(X, t)$ is a vector function that determines the rate of change of $X$ at each time point. Here, the equation represent the *flow* of the system, as we have a continuous system. Had it been discrete, we would have had a *map* instead.
The flow is like a river with a current : everything that we put in the river will follow the current, sometimes with some topological specificity. Putting a small wooden branch on the river is like starting a trajectory from an initial condition : from it, the branch will create a trajectory, going with the current. If we put another branch on another initial condition, its trajectory will have similarities and differences with the other one, unless the flow is completely uniform.

A common way to analyze dynamical systems is through the *phase space* (Strogatz, 2019). The phase space is a mathematical space representing all possible states of a system. It allows the visualization and analysis of the system's behavior and properties, such as stability and attractors. Each point in phase space corresponds to a specific state of the system at a given time. This global visualization is helpful to get a feeling of how the system can evolve, instead of following that evolution through time. Actually, if we follow enough trajectories in the phase space, we will start to be able to have a visual representation of the flow. Most of our analysis will be in the phase space, sometimes without a specific mention of it.

While a vast variety of dynamical systems categories exist, we will introduce a specific dichotomy : *conservative* and *dissipative* systems.
In conservative systems, the total energy remains constant over time. Energy can transfer between different components or variables within the system, but the overall energy of the system remains unchanged. In these systems, the trajectories remain inside of a fixed volume, although its shape can change. Classical examples include simple harmonic oscillators and ideal pendulums.
In contrast, dissipative systems are dynamic systems that lose energy over time. The system tends to approach a state of equilibrium or lower energy. Energy is dissipated, often in the form of heat, sound, or other forms of energy loss. In dissipative systems, trajectories in phase space tend to converge towards specific attractors, indicating the loss of energy over time. Examples of dissipative systems include damped oscillators, fluid dynamics with friction, and chemical reactions with energy dissipation.

As said here, conservative systems were the basics of the study of complex systems. However, being by definition closed and isolated, conservative systems offer little perspective on countless observable phenomena involving openness at their core, such as complex chemical reactions, turbulent behaviors, or living creatures. These phenomena would later be gathered under a key umbrella term : *dissipative structures*.
Although they have been at the core of a great part of modern investigations of dynamical systems theory, mostly through its chaotic branch, their dissipative nature and, more precisely, its formulation, has not

drawn much attention as a subject *per se*, out of strongly mathematical backgrounds (Pikovsky & Politi, 2016).

It is from these perspectives that dissipative systems have gained a lot of importance. These systems are the dynamical description of out-of-equilibrium thermodynamic systems, which embed "openness" in their core : they can exchange energy or matter with their environment. However, dissipative systems are poorly described and discussed in most courses and books dedicated to dynamical systems theory. On the other hand, many have heard about chaos and strange attractors, which are exactly the manifestation of a "stable" system out of equilibrium. But little is actually said on their dissipative nature, which has important consequences : it is what allows them to feature all these very entangled structures. Actually the mathematics literature on dynamical systems theory has taken these so-called Non-equilibrium Steady State (Ruelle, 2003) (the word stable should not be used here as chaotic systems are unstable in essence) as the center, the point of departure of further analyses.

Here, we will not dwell on those complex mathematical planes, as the goal of this thesis is not to give a proper mathematical formalism of the study of various neural networks. We will stop at an early stage of dynamical systems, using simpler tools, but it seems like a proper introduction was required, if only to get a glimpse of how deep one could go by studying neural networks from that approach.

**For computational neuroscience**
Let us now go back to our main object of analysis. Dynamical systems modelling has played a major role in the development of theoretical and computational neuroscience, covering a wide range of models, from single neurons (E. Izhikevich, 2004) to neural tissues (Wilson-Cowan (Wilson & Cowan, 1972)), and whole brain networks (TVB : The Virtual brain, as used in (Sanz-Leon, Knock, Spiegler, & Jirsa, 2015)). All of these models result from complex assemblies of experimental measures, observations, first principles, and even intuitions, in order to yield the best possible approximations of the object under investigation given the available means (being experimental or computational).
On top of the modeling, great analysis focus on thermodynamics and dynamical systems therory exist, such as (Cessac, 2019), which shows the potential power of that method when it comes to understanding computational neuroscience.

As said previously, modellers need to resort to some degree of phenomenology to represent neurons. This opens the door to a rather common, yet particular type of dynamical system that we introduced previously : dissipative systems.

In the computational neuroscience literature, the dissipative nature of phenomenological models is often little discussed, although it might give deep insights on the multi-scale nature of the dynamical processes involved : from such perspective, dissipation can be viewed as something "leaving" (or entering) the system, being matter, energy, or other, depending on the nature of the description involved. Here, a short note of importance is required : while the definition of energy is usually not that difficult mathematically speaking, a physical definition is often somewhat harder. Worse, we often introduce generalized energy that functionnaly serve the same role but are not correctly defined mathematically. While we will not explore the energy of the networks here, it is useful to remind us that it will be "pseudo energy" that the will be dissipated, and that the study of said energy would likely be of great interest to understand our dissipative networks better. As we said, a link with actual "physical" energy might sometimes be complicated to establish from a phenomenological model, but a somewhat more fundamental one exists with entropy defined from the phase space density : the dissipation of the system is directly related to the rate at which it *gives* entropy to the outside world. Here we fall back on the centrality of dissipative structures in Prigogine's perspective : being able to give entropy to the rest of the world, such a structure can maintain a more ordered structure without violating the laws of thermodynamics.
Entropy, while a fascinating subject, often revolves around the thermodynamic analysis, which takes a stochastich approach instead of the deterministic we chose here. There is no doubt that such an approach would be very interesting, and there is in fact a lot of works between the dynamics of the system and the entropy (Ruelle, 1996; Gallavotti, 2004; Gaspard, 2020).

## 3.2 Lorenz system

Dynamical systems are a complex subject, and it can be difficult to understand it when it is mixed with highly dimensional, non-trivial systems such as the simulation of neural networks we study here. It is also hard to know if the algorithms we want to use are actually working as we cannot predict the results

from said neural networks analysis. Therefore, to understand the concept and to test the algorithms, it is useful to use a simpler, well-known model from which results are already well-established. The model we choose to do that is a very famous one : the Lorenz system (Lorenz, 2004). It is a 3-dimensional system, so it is easy to represent it, it shows a variety of behavior depending on a change of some parameters, from fixed point attractor to chaos, and most of all it is well-known, eg in (Sparrow, 1982; Viana, 2000; Pchelintsev, 2014; Leonov, Kuznetsov, Korzhemanova, & Kusakin, 2015).
The system is defined as such :

$$F(\boldsymbol{X}) = \begin{cases} \dot{x} = \sigma(y - x) \\ \dot{y} = x(\rho - z) - y \\ \dot{z} = xy - \beta z \end{cases} \tag{14}$$

With $x, y, z$ the three dimensions and $\sigma = 10, \beta = \frac{8}{3}$ the classical parameters to have a chaotic system, with $\rho$ being variable to have different kinds of dynamical behaviors, from going to a fixed point if $\rho < 24.74$ to chaos if it is above, such as the typical system where $\rho = 28$.
Here is how the system looks like for various rho :

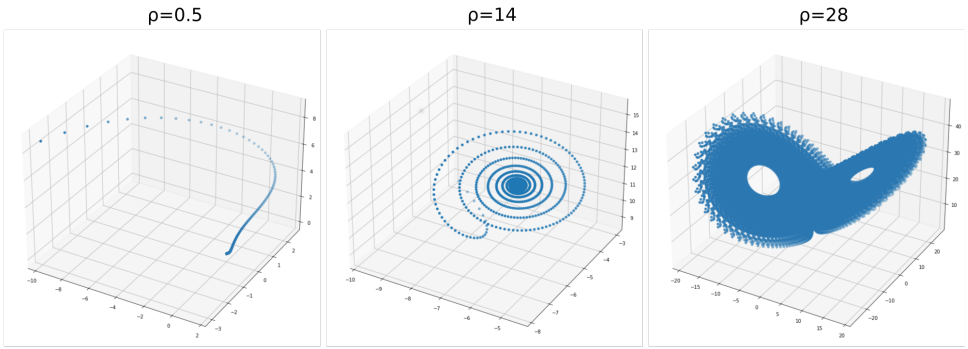


Figure 6: Illustration of the Lorenz system for three different values of $\rho$ showing three different behaviors, all simulations lasting for the same amount of time. For the first one, when $\rho < 1$, the dynamic goes straight to the $[0, 0, 0]$ point. When $1 < \rho < 24.74$, the dynamic goes to either of twin fixed points in a spiral way (only one of them is represented here). Finally, when $24.74 < \rho$, the system is (generally) chaotic.

## 3.3 Lyapunov exponents

### 3.3.1 General presentation

One of the main question asked about dynamical system is regarding their stability. Aleksandr Lyapunov was the first to formally adressed that question is his PhD thesis, in 1892, and gave his name to the main tool used to study said stability : Lyapunov Exponents (LE).
We will briefly introduce the LE, for a more detailled and heavily mathematical presnetation, please refer to (Eckmann & Ruelle, 1985), and for a more intuitive introduction, to (Pikovsky & Politi, 2016), which is the main source of inspiration for this part.

Let us consider a system of $N$ dimensions $\dot{\boldsymbol{X}} = F(\boldsymbol{X})$, a time-independant flow where $\boldsymbol{X}$ is a vector of $N$ dimensions, and $F$ a function.
As we question the stability of the system, we are interested in the evolution of an infinitely small perturbation from a point $\boldsymbol{x}_0$. Said evolution $\dot{\boldsymbol{x}}$ will follow

$$\dot{\boldsymbol{x}} = \frac{\partial F}{\partial \boldsymbol{X}}(t)\boldsymbol{x}(t) = J(\boldsymbol{X})\boldsymbol{x} \tag{15}$$

Where $J$ is called the Jacobian of the system and is defined as the matrix of the partial derivative of $F$ such that $J_{ij}(X) = \frac{\partial F_i}{\partial X_j}$.
Integrating this over $t$, we obtain

$$\boldsymbol{x}(t) = H(\boldsymbol{X}_0, t)\boldsymbol{x}(0) \tag{16}$$

with

$$H(\boldsymbol{X}_0, t) = exp(\int_0^t dt' J(\boldsymbol{X}(t')) \tag{17}$$

Finding the stability of eq.(16) is equivalent to knowing if, when $t \to \infty$, if $\boldsymbol{x}(t)$ grows or decays.
There is no reason to think $\boldsymbol{x}(t)$ will follow the same perturbations, thus we are actually interested in its norm to only know of the amplitude of the perturbation.

$$||\boldsymbol{x}(t)||^2 = ||H_t \boldsymbol{x}(0)||^2 = \boldsymbol{x}^T(0) H^T(t) H(t) \boldsymbol{x}(0) \tag{18}$$

where $\boldsymbol{x}^T, H^T$ are the transpose of $\boldsymbol{x}, H$.
We therefore only need to investigate the matrix $M(t) = H^T(t)H(t)$ to understand the evolution of the perturbation $\boldsymbol{x}(t)$. But does $M$ converges ? That question was answered in the seminal work of Oseledets in 1968 (Oseledets, 1968), discovering the so-called Oseledets theorem which proves that, under certain conditions, the following limit exists :

$$\lim_{x \to \infty} [M(t)]^{\frac{1}{2t}} = \Delta \tag{19}$$

and $\Delta$ is a matrix with $N$ positive real eigenvalues $\mu_i$.

Finally, we defined the Lyapunov exponents $\lambda$ as :

$$\lambda_i = ln(\mu_i) \tag{20}$$

Which can intuitively be interpreted as follow : if $\lambda_i$ is positive (respectively negative), it means the perturbation is growing (respectively decreasing)following the associated eigenvector.

We did not go into the serious mathematical details, but it is essential to note that ehe main condition to apply Osedelets theorem is that the system must be ergodic.
Ergodicity is one of the important part of dynamical system analysis, and a thorough presentation can be seen in (Eckmann & Ruelle, 1985). Here, we will simply say that an intuitive understanding of ergodicity is that a trajectory in the system will eventually visit all the points the system can visit. It can be translated into thinking that the mean value of the system following a trajectory from one initial condition for $t \to \infty$ is the same as the mean value from $n$ initial conditions, with $n \to \infty$.

### 3.3.2 A specific case : fixed points

Now that we have introduce the genral case of Lyapunov exponents, we can focus on a specific, simpler case to gain a more intuitive feeling of their usefulness and of the definition of stability : fixed points.
It is interesting to find points $\boldsymbol{X}^\star$ where there is no longer any evolution, e.g.: when $F(\boldsymbol{X}^\star) = 0$, the so-called fixed points.

Here, we can see we have a very specific case compared to the general one : $J(\boldsymbol{X}^\star)$ is now a constant, as $\boldsymbol{X}^\star$ will stay the same for all $t$. This means that eq.(17) is now trivially $H = exp(J(\boldsymbol{X}^\star)t)$. Hence, eq.(16) is now :

$$\boldsymbol{x}(t) = \boldsymbol{x}(0) e^{J(\boldsymbol{X}^\star)t}$$

If we diagonalize $J(\boldsymbol{X}^\star)$, we now have $N$ eigenvalues $\mu_i$ (possibly with an imaginary part) and $N$ corresponding eigenvectors $eig_i$. Taking the real parts of $\mu_i$ gives us $\lambda_i$ (as they are defined as reals), and finally :

$$x_i(t) = x_i(0) e^{\lambda_i t}$$

Thus, all that it required to know the stability of fixed points is to know the real part of the eigenvalues of the Jacobian matrix (and the eigenvectors if we want to know the direction of those (in)stabilities).

How do we interpret the LE we can now compute ?
If $\lambda < 0$, the perturbation will converge to the fixed point, said fixed point is called an attractor. If $\lambda > 0$, the perturbation will go away from the fixed point (it is also called a source). More than one

exponent can show us the same thing for different directions, either being all positive or negative, or being a mixture of both. The magnitude of $\lambda$ will show how fast the system converges or diverges. As they are related to an exponential term, the other LE after the first one will have less of an impact on the dynamic, but they will still be relevant regarding their eigenvector direction, and will influence the global dynamics in a chaotic system.

It is easy to represent those by thinking of a valley or a mountain. A point in the middle of the valley is stable : even if you move it, it will come back to its initial position. A point on the top of the mountain is unstable : by moving it a bit it will fall. At 2D, the same example applies if the 2 directions are stable or unstable. If it's a mix, we have a so called saddle point. All three cases are represented in Fig.7

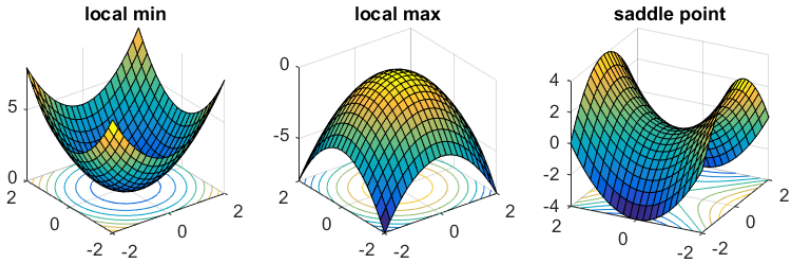


Figure 7: Illustrations of the different LE values from https://www.offconvex.org/2016/03/22/saddlepoints/. On the left, we have a local minimum, corresponding to all negative LE. In the middle, we have a local maximum, corresponding to all positive LE. On the right, we have a saddle point : one LE is negative (in the North-West/South-East axis) and the other one is positive (in the North-East/South-West axis).

This representation is identical as seeing how a local perturbation around the fixed point would evolve.

### 3.3.3 An example with the classical Lorenz system

In this part, we aim to apply the Lyapunov exponent analysis to the Lorenz system (Eq.(14), where $N = 3$) in order to give a better understanding of its dynamics.
The easy, but nevertheless interesting way to analyse such system is to find their fixed points, as defined previously.
Here, we can see trivially that one such a point is $(0, 0, 0)$. Resolving the equations, we can also identify two other points : $(\sqrt{\beta(\rho-1)}, \sqrt{\beta(\rho-1)}, \rho-1)$ and $(-\sqrt{\beta(\rho-1)}, -\sqrt{\beta(\rho-1)}, \rho-1)$.

Now, we need to learn what happens around those points, to see if the dynamics will go toward them, away from them, or a mixture of both. To do so, we linearized the equation around them, computing the Jacobian $J$ of the system.

$$J = \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho - z & -1 & -x \\ y & x & -\beta \end{pmatrix} \tag{21}$$

Replacing $(x, y, z)$ with the values of the fixed points, we can obtain the linearized behavior of the system around said points.
To know how they behave, we have to compute the eigenvalues of the matrix while diagonalizing it. The real part of said eigenvalues will be the previously defined LE, ordered from the biggest to the lowest exponents.

For the fixed points listed before, with the classical parameters, we have the (approximate) following LE : $(11.83, -2.67, -22.83)$ for (0,0,0) and $(0.094, 0.094, -13.85)$ for the other two symmetric point.
All those points have a positive first lyapunov exponent. Let's focus on each of them to understand correctly what they mean.

For (0,0,0), the first exponent is strongly positive, meaning the flow will get quickly away from it in one direction. The next 2 exponents are negatives, so there are 2 attractive directions in the phase space, and when the trajectory is close enough to (0,0,0) it will get quickly away from it. All of them are only real, so the trajectory will go directly toward or away from the points.
For the other 2 symmetric points, the first 2 exponents are weakly positive with an imaginary part, which means the trajectory will slowly go away from the points in "spirals". The last exponent is strongly negative and real, so there is a direction of attraction that goes straight to the points. This means that

around the points, the trajectory will slowly spiral away of it, before arriving to the direction that pull the trajectory strongly toward it, making said trajectory "turn around" the fixed point for a time before leaving to go around the other symmetric point, which is the observed dynamics we see in Fig.6(right).

We can also infer that the dynamics is chaotic : there is no global attractor, (with all negative LE), so the system will just follow the flow toward some specific direction, going closer to some fixed point, then be rejected outside, and continue like that. As we know the system is bounded (and is, in fact, dissipative, as we will prove later), it means it is chaotic (although actually it could also be periodic, but we will prove it is chaotic on the next part).

The LE helped us do an easy analysis of the behavior of the system and understand how it worked locally, which gave us precious understanding around those three points, with good extrapolations on the global dynamics, on top of the high suspicion of its chaotic nature.

### 3.3.4 Going deeper : using LE for the system instead of fixed points

We have used the simple case of the fixed points to get a feeling of how the LE can be used, but what about the rest of the system ? Following the general definition we gave earlier, we see that it is possible to find a converging LE from (almost) any starting points. Here we will briefly defined the specific algorithm that we will use, before we assess its validity.

The main idea behind the algorithm is to use the so-called QR decomposition. To do so we compute the evolution of a vector in the tangent space and to decompose it. The main direction of evolution when the evolution is long enough is due to the first LE. Then we impose that the second vector is orthogonal to this one, in a hyper-plane. We force an orthonormalization through the Gram-Schmidt process, and obtain the second vector. We keep doing as such, going to lower subspace, until we have created N exponents. It is also possible to obtain the vectors, the CLV (Covariant Lyapunov Vectors) by using a more complicated algorithm, but we will not detail it here. Note that those algorithms require to have the equations of the flow. For more details, we can refer to (Pikovsky & Politi, 2016).

While difficult, it is possible to do it for simple systems such as the Lorenz system we have been studying. It is possible to use various parameters to obtain good results in a reasonable time, which allow to have, for the classic system, what can be seen in Fig.8

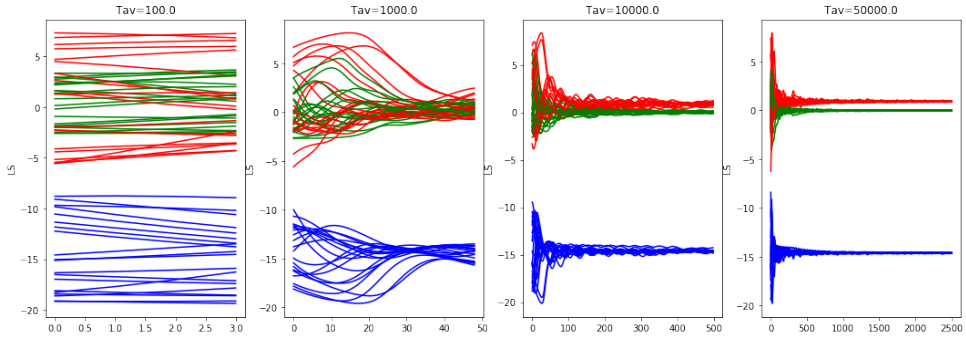


Figure 8: LE of the Lorenz system obtained from the Politi algorithm. Each panel shows a different time of evolution in the tangent space (1000,50 000 increments) of 20 different origins taken randomly. The colors represent the 3 LE : red is the first one, green the second one, blue the third one. When the time of evolution is long enough, it converges to specific values.

Applying the algorithm, we find the values for the 3 system LE as follow $[0.905, -8.26e-05, -14.572]$ : the first one is positive, indicating chaos (for sure this time), the second one is actually 0, and correspond to the direction of the flow (it always appears for dynamical systems as there is always a direction that can follow the flow) and the last one is highly negative, being responsible of the dissipative nature of the system. To interpret it a bit more, going back to the exponential representations we showed earlier, it means that on average, if we take a random point of the system and perturb it, the distance between the initial point and the perturb one will grow to infinity. This is the so-called sensitivity to initial conditions, which is a marker of chaos. Here, one can observe that as the system is bounded, the perturbations will actually not grow to infinity. Actually, after some time of growing apart, they will start getting closer,

then farther, without any specific pattern. This is because the linearization is only valid at the proximity of any points that is used, and is technically invalid as soon as we are away from it. It is a representation of what would happen if there were no other modifications of flow around, which is why we talk of a tangent space : a space that only touches the system locally and simplifies it for the sake of finding what happens locally. Actually, more than a growing distance, what happens is that there is a growing decorrelation between the two initially neighboring trajectory, however close they were to begin with.

As can be seen on the first panel of Fig.8, there are lines of the FLE that are sometimes below the one of the second LE. This is due to the nature of the algorithm : we impose a direction corresponding to the FLE, and while it is often true, the direction also shifts sometimes faster than what the algorithm can predict, which is why we can have a mixture of a close LE at the beginning. This should be a good reminder as to why we need to wait for convergence (as in the second panel).

### 3.3.5 Computing LE from time-series in the Lorenz system

In the previous parts, we were able to use the equations of the system. It is obviously impossible from experimental data. Worst, they are always partial whether it is because we are recording only some dimensions (like the membrane potential) and not others (like quantum effect at lower scale or the effect of astrocytes) or because we are only recording some neurons out of the whole.
Often, when the goal is to measure an evolving activity, we have access to time series : partial data that changes through time and aims to represent the whole. Time series have been studied a lot in general (Broomhead & Jones, 1989; Härdle, Lütkepohl, & Chen, 1997; Fu, 2011) and within dynamical systems (Packard, Crutchfield, Farmer, & Shaw, 1980; Yu, Chen, Cao, Lü, & Parlitz, 2007), more specifically, the question of how to use them as a medium to an inaccessible whole has been asked and some answers have emerged. One of those answers is embedding, which aims to reproduce the whole many dimensional activity for a one-dimensional time series (Strogatz, 2019).

**Takens' theorem** Takens' theorem, proposed by Floris Takens (Takens, 1981), states that it is possible to reconstruct the underlying dynamics of a system from a single observable time series using a technique called time delay embedding.The idea is to create a higher-dimensional space through the so called time delay embedding.

The time delay embedding is a technique used to reconstruct the underlying dynamics of a system from a single observable time series, $x(t)$. It involves creating a higher-dimensional space called the embedding space - and a higher dimensional corresponding vector $X(t)$ - by embedding the time series with delayed copies of itself. The creation of the new vector is done this way :

$$X(t) = [x(t), x(t-\tau), x(t-2\tau, ..., x(t-(m-1)\tau] \tag{22}$$

Here, $X(t)$ is the $m$-dimensional vector in the embedding space at time $t$, and $\tau$ represents the time delay between successive copies of the time series. The parameter $m$ is called the embedding dimension.
By selecting appropriate $m$ and $\tau$, the reconstructed state space preserves the essential dynamics of the original system. This enables the analysis and prediction of complex systems, even when only one observable variable is available.

Now that we have introduced how it is possible to reconstruct a system from time series, we will show an application on the Lorenz system, as in Fig.9. We use an algorithm that takes care of the question of the embedding dimension, greatly inspired from https://www.kaggle.com/code/tigurius/introduction-to-taken-s-embedding.

To do so, we simulated the system, then took the simulated points (making them dense enough by doing the simulation for a long enough time), choose a dimension (x,y or z) and use the points from that dimension : this is our time series. We do not detail the process of the algorithm or the found delays and embedding dimension, only the 3D representation that results from the reconstruction in Fig.9.

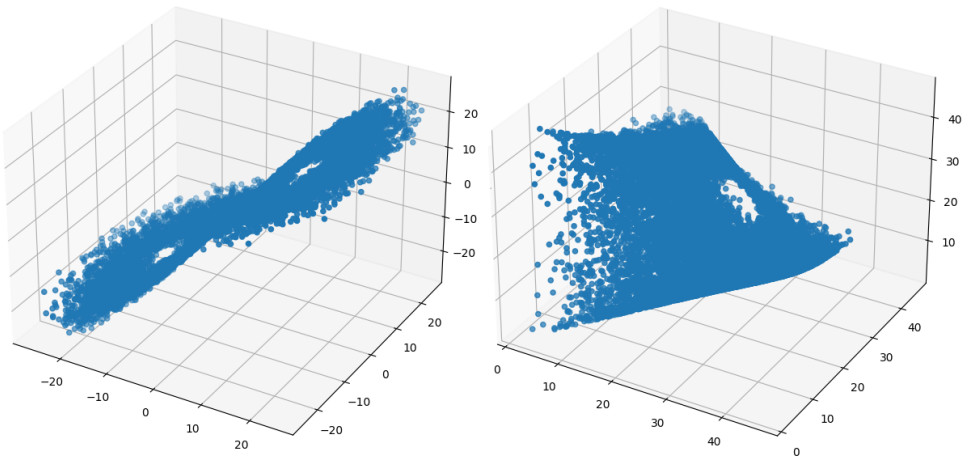


Figure 9: Illustration of the reconstructed Lorenz system from Takens' theorem. On the left, we have a pretty good reconstruction, following the original topology. This is the reconstruction obtained from the first or second dimensions. On right, we have a not so good reconstruction obtained from the third dimension

As can be seen in the figure, the algorithm doesn't seem to "work well" for all dimensions. We can see that the two wings of the butterfly seem to be folded together in the right part of Fig.9. This is an interesting analogy considering Takens' theorem is supposed to "unfold" the dynamics.
But this result makes sense when we look at the equations of the Lorenz system : we can see that the third dimension is symmetric when it comes to the other two, which means that we cannot completely separate all the dimensions properly.

Most of the time, unless there are some specific symmetric issues as we have here (that also arise because of the simplicity of the system), the reconstruction goes well, and it always gives meaningful information.

We now focus on the left reconstructed system to go further.

**Computing the LE from the new system** Then, when the new system is created through the method of Takens' theorem (or while imposing the dimensions and their associated variables, which we did in chapter 3), we can compute the LE.
The details are given in chapter 3, the core idea being that we can artificially "perturb" the system by finding nearest neighbors and follow the evolution of the two trajectories (the original one and the "perturb" one) to compute the exponent (Wolf, Swift, Swinney, & Vastano, 1985; Eckmann, Kamphorst, Ruelle, & Ciliberto, 1987). The details of the code are given in the annex.

We applied it and found Fig.10, with a FLE converging to 0.897, a fairly similar value to the previous one at 0.905. This result is interesting because, as the FLE is positive, it already allows us to claim that the system is chaotic. This means even one of the dimensions already contains the information we need to see if the system is chaotic or not, which is generally the main goal of such time series algorithms.

Note that, as we will detail in the appropriate chapter, this algorithm is very good when it comes to finding positive exponents, but not so good for the rest. Knowing the $2^{nd}$ LE was close to 0, we did not try to compute it with this method and stopped at the FLE which was a success and allowed us to properly test our algorithm.

### 3.3.6 Summary of Lyapunov Exponents

We have explained how the LE worked, how it was possible to use them on fixed points, on a whole system, and from time-series. To illustrate those points, we used a classical example : the Lorenz system, and reproduced well known results for the classical chaotic parameters representing the famous "butterfly" shape. We presented the 3 fixed points, analyzed their stability, and explained how the system was chaotic. Then we reproduce that chaotic result with only some partial data and without the equations, using time-series algorithms. We will use that specific analysis in chapter 3.

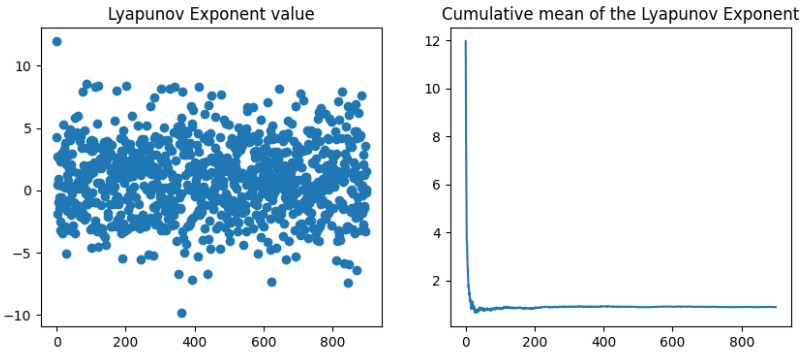


Figure 10: Applying the FLE algorithm on the reconstructed Lorenz system. On the left, we have the values of the exponents at various steps. On the right, we have the cumulative mean. The algorithm converges to 0.897

But while powerful, there is an issue with the Lyapunov analysis : it revolve in diagonalizing a matrix of the equations, or using nearest neighbors with time-series. That works well for the Lorenz system, as we have seen, but the story changes completely in high dimension, partly due to the so-called curse of dimensionality (Köppen, 2000), partly because diagonalizing becomes extremely hard to operate. To provide a solution to this issue, we will now present the dissipation.

## 3.4 A solution to analyze the dynamics in high dimensions : introducing the dissipation

### 3.4.1 A mathematical description

Mathematically speaking, the dissipation relates to the evolution of volumes of initial conditions in phase space. For a general system whose evolution equation can be written in the form:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t)) \tag{23}$$

where $\mathbf{x}(t)$ is the state vector, we are concerned with the time evolution of a volume $\mathcal{V}$, representing a continuum of initial conditions, delimited by a closed surface $S$, with outward normal vector $\mathbf{n}$:

$$\frac{d\mathcal{V}}{dt} = \int_S \mathbf{F}(\mathbf{x}) \cdot \mathbf{n} \; dA \tag{24}$$

where $dA$ is an infinitesimal surface element taken on $S$. By the divergence theorem, it is straightforward to obtain

$$\dot{\mathcal{V}} = \int_V \nabla \cdot \mathbf{F}(\mathbf{x}) \; dV \tag{25}$$

Hence the time evolution of the volume enclosed in the closed surface $S$ is given by the divergence of the evolution function integrated over the same volume. Volume preserving dynamical systems, such that $\dot{\mathcal{V}} = 0$ are called conservative.

A system for which $\mathbf{F}(\mathbf{x}) \cdot \mathbf{n} < 0$ in some region of phase space is called dissipative. Interestingly, shrinking of volumes in phase space typically implies the existence of *attractors*, which (loosely speaking) are bounded subsets to which nonzero volumes in phase space asymptotically converge.
Therefore, we will specifically study that part, that we define as :

$$Diss = \nabla \cdot \mathbf{F} \tag{26}$$

It is interesting to note that a negative global dissipation means that the system will keep losing volume, and one could easily think the trajectories will therefore all converge to a volume of 0, which seems a bit strange. Intuitively, it would seem that all go to a point, which would not be a very interesting dynamic.

While this is a possibility, it is not the only one. In fact, the volume is define on a specific space, and one has to remember that every measure on subspaces will have a volume of 0 e.g : a surface in 2D and a line in 1D, if put in a 3D space, have a volume of 0, the same is true for any measures on a subspace of dimension $m - i$ in a dimension $m$, for all $0 < i < m$. This therefore means that the trajectories will converge to a specific subspace of dimension lower than the original one. Without going into details, that is how fractal works : objects of non integer dimension, which is, for example, the case of the butterfly shape of the Lorenz system (estimated to be around 2.06(Kuznetsov, Mokaev, Kuznetsova, & Kudryashova, 2020)).
This means that dissipative systems can dissipate without their dynamics being reduced to mere uninteresting points.

### 3.4.2 Link with LE

Dissipation is related to the evolution of a volume of points within the flow of the phase space. It is, of course, related to the LE we presented earlier, and that does the same thing with more details, specifying how directions evolve. We can see dissipation as a rough vision of LE, giving only the general sense of stability, showing if we have a dissipative system or not, and if we do, how fast it dissipates. This is why it is not surprising to know those two objects are linked : the dissipation is actually the sum of the LE.

$$Diss = \sum \lambda_i \tag{27}$$

Here is a rough proof, which also helps to understand what dissipation does :

- LE are defined as the real part of the eigenvalues of the Jacobian.
- The trace of matrices is the sum of eigenvalues.
- Therefore the trace of the Jacobian is the sum of all LE
- Therefore, the sum of LE is equivalent to the partial derivative of all dimensions of the system, relatively to said dimension
- Therefore, the sum of LE is equivalent to the divergence of $F$
- Dissipation is the integral over a volume of the divergence of $F$
- Integrating on a volume in the phase space, dissipation is equivalent to the sum of LE.

Obviously, the sum of the LE gives way less information than the LE themselves, we only know the general trend on how stable the system (or eventually sub-system) is, which can be translated by how fast a the flow goes back to the attractor after a perturbation.
On the other hand, as explained before, it is not reasonable to try to obtain the LE in very high dimension, while this problem is no longer an issue with the dissipation. This explains why we focused here on the dissipation: while being less precise than the LE, it is impossible to use them, and still possible to learn a lot from the dissipation.

### 3.4.3 A feeling of dissipation : understanding with Lorenz

We come back to the Lorenz system to get a better feeling of what the dissipation means. Of course, as we explained before, the LE analysis is perfect for low dimensions systems, and we actually already learned a lot about the Lorenz system. Using the dissipation will make us learn new things, but, as it is the sum of the LE and we already computed them all, nothing that we couldn't have known if we dug a bit more what the LE could teach us. We therefore use this mostly as a way to understand what the dissipation is : while it is not the best tool for such a simple system, it will be very useful in chapter 4.

As presented before, dissipation is the rate of convergence of the flow. This should mean that the higher the dissipation (meaning, the farther from 0), the faster the convergence.
First, we can compute the dissipation of the system. From eq.21, we remember the Jacobian of the system, and using eq.27, we find that the dissipation for the Lorenz system is :

$$Diss_{lorenz} = -\sigma - 1 - \beta \tag{28}$$

Which, with the typical values of $\sigma$ and $\beta$ gives:

$$Diss_{lorenz} = -13.7$$

As we can see, the dissipation of the Lorenz system is negative. This was the characteristic of dissipative system we talked about previously : if we start a cloud of initial conditions and let the trajectories evolve, they will converge to a subspace of the attractor (the butterfly shape), losing volume, as we presented before.
It also has the interesting property of not depending on the position : it will remain identical at any points for a given set of parameters. More interesting, we have seen before that various $\rho$ leads to various behaviors, but the dissipation is independent of $\rho$. This is interesting, as it is already telling us that 1) Various behaviors can have an identical dissipation, and 2) that we should observe some similarities in the dynamic of all simulations of the Lorenz system in the way they converge to an equilibrium, even if said behavior is different.

We test that, initializing a cloud of points, and letting them follow the flow. We repeat that for various values of $\rho$. In order to measure a convergence rate, we compute the volume of the cloud of points using the Convex Hull.
It is important to stress a limitation of this method : while it can compute a volume easily when it is in 3D, the volume technically converge to 0 as the system goes to a subspace (a point if $\rho$ is low enough, and a fractal dimension $<3$ if $\rho$ is high enough). As there will be small variation, the function cannot really find a volume of 0 when there is a convergence to a subspace, which is what we find.
On top of that, some statistical outliers (although not random, the chaoticity of the system can lead to far from the average points) could drastically change the volume, which is why we remove the top 20% extreme points.
We can see the results in Fig.11:

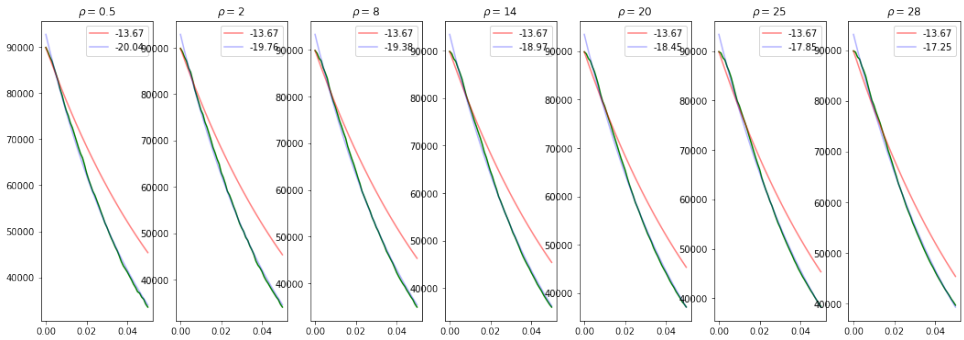


Figure 11: Evolution of an initial volume of points with time, zoomed to observe the differences. In red, we have the prediction curve, using the value of the dissipation. In green, we have the actual volume of points, from 8000 points. In blue, finally, we fitted the green curve to $f(x) = e^{D*t}$ where D is written. We repeat the process for various values of $\rho$.

As we have said the value of the dissipation and predicted rate of contraction is always the same for various $\rho$. We can see that the superposition appears to be fairly good. The value of the dissipation is therefore a good predictor of the time it takes for a system to converge to the equilibrium.

We also wanted to test the same situation with a perturbation. To do so, we let the system evolve until the equilibrium is achieved, and then we perturb it in many different ways, following with many simulations from each of the perturb points, which gives a cloud of points as previously. We have a similar result as before.

We also tried to check the time it takes for a perturbation to go back to the original equilibrium. To do so, we did the same as previously and defined some borders corresponding to what we considered to be "the space of the attractor". When a point re-enter this space, we consider it is back to the equilibrium. We measure how long it took to go back, as a function of how strong the perturbation was (ergo, how far was the perturb point from the original point). The goal here was to link that time to the dissipation, which we did coarsely, as it is only for the sake of showing how dissipation works.

We know that a volume evolves with $V(t) = V0e^{Dt}$ where $D$ is the dissipation. Therefore, the time $t$ to go to $V1$, a volume within the stable volume is $t = 1/Dln\frac{V1}{V0}$. We define $V1$ as the volume within

the borders we used previously to estimate when a perturb trajectory would be back to a stable system. A crude approximation of $V0$ is $d^3$, where $d$ is the distance between the original and the perturb point. This means we can reduce the previous form to : $t = aln(x)+b$, where $a = \frac{-1}{D}$, $x = d^3$ and $b = \frac{1}{D}ln(V1)$. We can see the result of the simulation if Fig.12

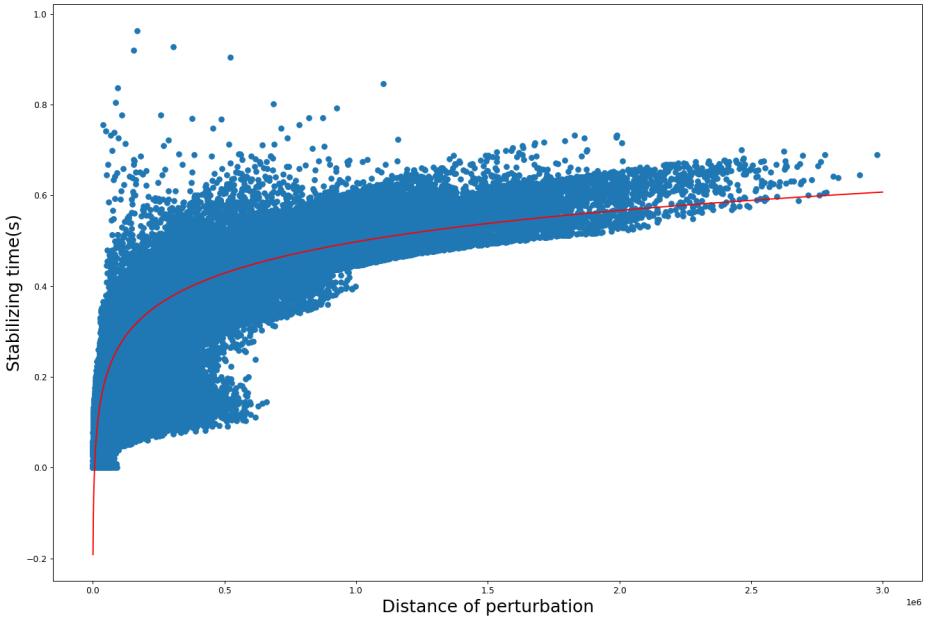


Figure 12: Lorenz system with $\rho = 28$. Time to come back to the stable system for various perturb points as a function of the distance of perturbation (blue) fitted (red) by a function $t = aln(x) + b$ where $t$ is the time, $d$ the distance and $a$,$b$ parameters.

From the theoretical values of $D$ and $V1$, and the fit, we obtain :
$a_{theory} \approx 0.0732$, $a_{fit} = 0.0998$.
$b_{theory} \approx -0.760$, $b_{fit} = -0.881$.

We can see that they are fairly similar despite the crude approximation we used. The dissipation can therefore indeed be used to compute the time it takes for a perturb trajectory to go back to the stable one.

### 3.4.4 Summary of the dissipation

The dissipation is the sum of all LE, but is very easy to compute as long as we have access to the equations of the flow, even in very high dimensions as it will just be a sum of more values. It gives us less information than the LE, and we can not determine the chaoticity of a system from it, but it still shows us directly how the system "recovers" from a perturbation. It is also a macroscopic measure that gives us information of the dynamics of the system in, as we said, an easy way. We will see in the later chapters that, due to its simplicity, it can also be used to unravel differences between networks.

As a side note, the power of the dissipation really lies on the simplicity to compute it with the equation of the flow. Conversely, it means the dissipation is not very useful with time-series are, more generally, experiment data. To the best of our knowledge, the only way to compute it would be through the sum of the LE using the method we presented earlier which gives, as we said, fairly bad results for negative exponents. This means it is unlikely that we could use the dissipation on real data.

# 4 What did we learn, and what will we need ?

This concludes our introduction, all the required knowledge and tools should have been presented so that it is possible to understand the next three chapters that will provide some results to understand the differences between apparently similar neural networks.

First, we have seen why we model reality, and we provided some warnings and urged to be careful about the way we treat those representations, explaining why it was required to pay more attention to those representations themselves in order to clearly differentiate those who could appear similar, and to select the appropriate ones when needed.

Then, we started to explain our object of analysis : the biologically realistic neural networks. After presenting how single neurons work, and mainly how we could model them using two different models (AdEx and HH), we explained how those single neurons could be connected to give a network. We explained some specificity of those networks, and presented the ones we will use in our work. We also showed why they were complex and why their analysis was difficult, and we presented the difference between Driven and Self-sustained networks, which, with the difference between single neuron models, will be the two main dichotomies we will explore. Finally, we presented a few computational tools that we will use : the representation of the network as a mean-field (taking a specific approach developed by the team), and the perturbations of the network.
We want to point out here that "models" might be used for two different situations : modelling in general, as in the very beginning of this introduction, which is similar to "finding a way to represent some object", and the single neuron models. We will most likely talk of the latter in the next chapters, and try to use "representation" when we want to talk of the former to avoid confusion.

After that presentation on the object we will analyze, we introduce the angle we will use to analyze them : dynamical system theory. We presented the concept of dynamical system theory, explaining how the differential equations gave us the flow in which each trajectories could evolve, and presented the so-called phase-space, which we will use implicitly in the next chapters. We then presented two important tools, starting with the Lyapunov Exponents (LE), explaining how they could help us understand local and global stability, and how to compute them, including from time-series (we will mainly use them in chapter 3). Finally, we introduced the dissipation, the sum of the LE, showed why it was a useful alternative on high dimensional systems (such as the networks we will use), and what it was measuring (we will mainly use them in chapter 4).

# Chapter 2 : Propagation or non-propagation of a perturbation


## Summary

Our computational study investigates how epileptic activity invades normal brain tissue, and shows the specific role of the inhibitory population, and its dynamical and structural aspects, using three different neuronal networks. We find that both structural and dynamic aspects are important to determine whether seizure activity invades the network. We also find that, despite those determinants, a part of the reason behind the effect remains random, hinting for complex dynamics we do not fully master. We show the existence of a specific time window favorable to the reversal of the seizure propagation by appropriate stimuli.


In this work, I participated in the simulations, analysis and writing parts.


Publication : (Depannemaecker et al., 2022), in eNeuro:
https://www.eneuro.org/content/9/6/eneuro.0234-21.2022

# A Model for the Propagation of Seizure Activity in Normal Brain Tissue

Damien Depannemaecker[1,*], Mallory Carlu[1,*], Jules Bouté[1], Alain Destexhe[1]
[1] Paris-Saclay University, French National Centre for Scientific Research (CNRS), Institute of Neuroscience (NeuroPSI), 91198 Gif sur Yvette, France
* equally contributing first authors

**Abstract**

Epilepsies are characterized by paroxysmal electrophysiological events and seizures, which can propagate across the brain. One of the main unsolved questions in epilepsy is how epileptic activity can invade normal tissue and thus propagate across the brain. To investigate this question, we consider three computational models at the neural network scale to study the underlying dynamics of seizure propagation, understand which specific features play a role, and relate them to clinical or experimental observations. We consider both the internal connectivity structure between neurons and the input properties in our characterization. We show that a paroxysmal input is sometimes controlled by the network while in other instances, it can lead the network activity to itself produce paroxysmal activity, and thus will further propagate to efferent networks. We further show how the details of the network architecture are essential to determine this switch to a seizure-like regime. We investigated the nature of the instability involved and in particular found a central role for the inhibitory connectivity. We propose a probabilistic approach to the propagative/non-propagative scenarios, which may serve as a guide to control the seizure by using appropriate stimuli.

**Significance:** Our computational study investigates how epileptic activity invades normal brain tissue, and shows the specific role of the inhibitory population, and its dynamical and structural aspects, using three different neuronal networks. We find that both structural and dynamic aspects are important to determine whether seizure activity invades the network. We show the existence of a specific time window favorable to the reversal of the seizure propagation by appropriate stimuli.

## 5 Introduction

Epilepsy is one of the most common neurological disorders (Beghi, 2019), which can take numerous forms. It is associated with the presence of paroxysmal electrophysiological events and seizures, usually recorded in humans using the electroencephalogram (EEG). However, EEG recordings do not allow us to probe the activity of single neurons within the network. More recently, the recording carried out with microelectrode arrays made it possible to obtain spike information of the order of a hundred neurons in human epileptic patients (Peyrache et al., 2012; Dehghani, Peyrache, Telenczuk, Quyen, et al., 2016; Paulk et al., 2022).

Such microelectrode recordings showed that neuronal activity during seizures does not necessarily correspond to synchronized spikes over the whole neuron population, as previously modeled (Soltesz & Staley, 2008), including models at different scales from cellular to whole-brain levels (Depannemaecker, Destexhe, Jirsa, & Bernard, 2021; Depannemaecker et al., 2020). In fact, it turns out that the dynamics of neural networks during seizures are more complex (Jiruska et al., 2013), and still poorly understood.

In particular, it is not known how the paroxysmal activity of the seizure does propagate, driving other networks into seizure activity.

Here, we investigate this problem using computational models.As a starting point, we consider examples of seizures where the inhibitory network is strongly recruited, while excitatory cells' firing is diminished. Fig.13 shows three seizures from a patient which were recorded using Utah-arrays, before resection surgery in a case of untractable epilepsy. From these intracranial recordings, 92 neurons have been identified and isolated and were classified into two groups: Fast-Spiking (FS) neurons and Regular-Spiking (RS) neurons, based on spike shape, autocorrelograms, firing rates and cell-to-cell interactions (Peyrache et al., 2012). Remarkably, direct cell-to-cell functional interactions were observed, which demonstrated that at least some of the FS cells are inhibitory while at least some of the RS cells are excitatory (see details in (Peyrache et al., 2012)). The three seizures shown in Fig.13 were taken from the analysis of (Dehghani, Peyrache, Telenczuk, Quyen, et al., 2016) (see this paper for details), and are shown with the firing rate of each population of cells. During the seizure, we can observe a plateau of high activity of FS cells, and a strongly reduced activity of RS cells. This phenomenon of unbalanced dynamics between RS and FS cells was only seen during seizures in this patient (Dehghani, Peyrache, Telenczuk, Quyen, et al., 2016). It shows that, in these three examples, the seizure was manifested by a strong "control" by the inhibitory FS cells, which almost silenced excitatory RS cells.

Interestingly, a very different conclusion would have been reached if no discrimination between RS and FS cells were performed, which underlies the importance of discriminating RS and FS cells for a correct interpretation of the dynamics during seizures. Based on such measurements, we built computational models based on larger number of cells in order to consider network effect that are not directly accessible with the recordings. We were interested in how seizure activity propagates or not, and what are the determinants of such propagation.

The region of the brain where the seizure starts is called the seizure focus, although in certain patients it can be distributed over several foci (Nadler & Spencer, 2014), then the seizure spreads to other regions of the brain. When another such region is reached, it can in turn be driven into seizure activity, in which case the seizure activity propagates. It can also control it (as in Fig.13, which was done with data from (Dehghani, Peyrache, Telenczuk, Quyen, et al., 2016)), in which case the seizure would remain confined to a more restricted brain region.

In order to gain understanding of the dynamics underlying hese two scenarios, we study the response of networks using three different neuron models (Adaptive exponential Integrate and fire (AdEx), Conductance-based Adaptive Exponential integrate-and-fire (CAdEx), and Hodgkin-Huxley (HH) models), interacting through conductance-based synapses, to an incoming paroxysmal (seizure-like) perturbation. We observe two types of behavior which we represent in Fig.14: one where the incoming perturbation successfully

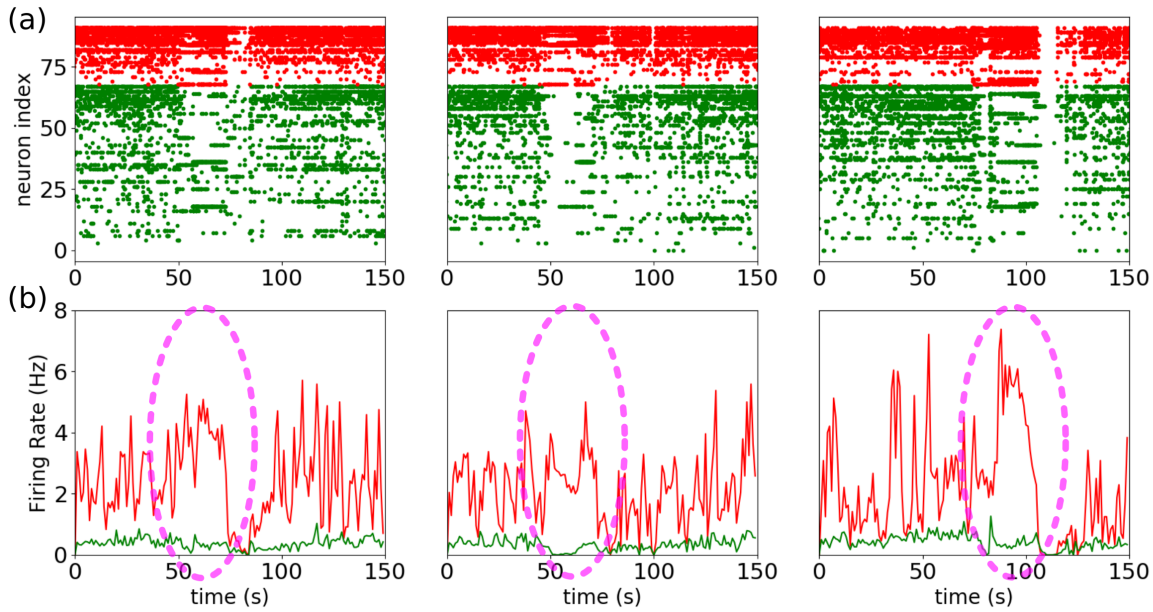


Figure 13: Examples of inhibitory recruitment during seizures. **a**, Raster plot of three different seizures from the same patient, 92 neurons were identified, 24 putative inhibitory cells (red) and 68 putative excitatory cells (green). **a**, Corresponding firing rate of the putative inhibitory population (red) and the putative excitatory population (green). A plateau of high activity of the putative inhibitory cells can be observed during the seizure (highlighted in dashed purple oval). This was done with data from the study by (Dehghani, Peyrache, Telenczuk, Quyen, et al., 2016)

increases the activity of the excitatory population, thus making it stronger than the input, and the other where only the inhibitory population strongly increases its activity, thus controlling the perturbation. In the first case, where the excitatory population discharges very strongly, it is therefore likely to transmit, or even amplify the perturbation transmitted to the next cortical column . We have therefore called this situation the propagative scenario. In the opposite case, where the firing rate of the excitatory population remains much lower than the perturbation, the seizure-like event will not spread to the neighboring region, we therefore call this situation the non-propagative scenario. We then propose a more precise approach, based on the AdEx network, that mixes structural and dynamical ingredients in order to unravel key aspects of the mechanisms at play. Focusing on the different input connectivity profiles for each node in the network, we are able to build separate groups of neurons that display significantly different dynamics with respect to the perturbation. Finally, we study the possibility of a proactive approach, based on the application of an extra stimulus with the aim of reversing the propagative behavior, thus controlling the spread of the seizure.

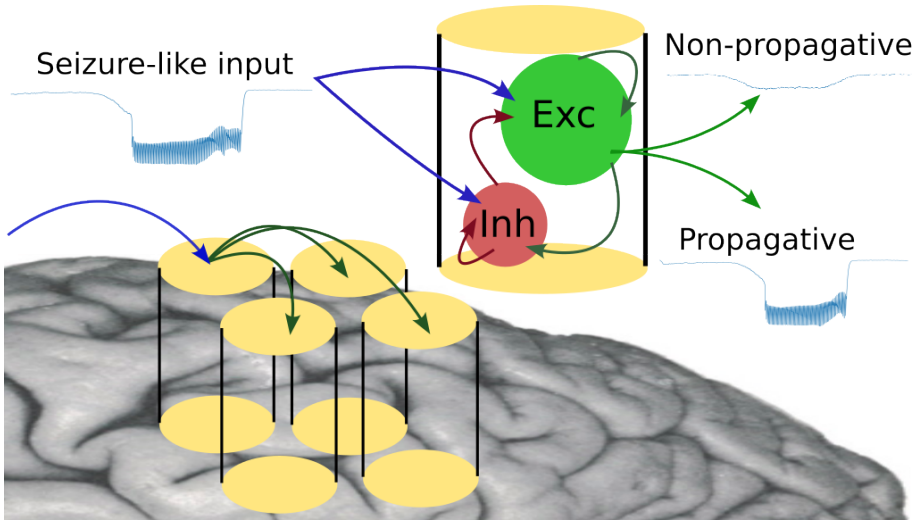


Figure 14: **Cartoon of the modeled scenarios**

# 6 Material and methods

## 6.1 Computational models

We use for this study a mathematical model of electrophysiological activity based on ordinary differential equations, describing the dynamics of the neurons' membrane potential through their interactions.

Each neuron model in the network is described by Eq.(29) and Eq.(30), the Adaptative Exponential integrate and fire (AdEx) model (Brette & Gerstner, 2005b; Naud, Marcille, Clopath, & Gerstner, 2008b).

$$C\frac{dV}{dt} = g_L(E_L - V) + g_L\Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) - w + I_{syn} \tag{29}$$
$$\tau_w \frac{dw}{dt} = a(V - E_L) - w$$

When the membrane potential crosses a threshold, a spike is emitted, and the system is reset:

$$\text{if } V \geq V_D \quad \text{then} \begin{cases} V \to V_R \\ w \to w + b \end{cases} \tag{30}$$

Parameters used for the excitatory (RS) and inhibitory (FS) populations are respectively $V_t = -50$ mV and $V_t = -48$ mV, $\Delta_T = 2$ mV and $\Delta_T = 0.5$ mV, $b = 100$ pA and $b = 0$ pA, and $\tau_w = 1000$ ms for RS. For both populations: $C = 200$ pF, $g_l = 10$ nS, $E_l = -65$ mV, $a = 0$ nS, $V_{reset} = -65$ mV, $t_{refractory} = 5$ ms.

In order to compare some of the results obtained with the AdEx model we used two other models of

neuronal activity. First the Conductance-based Adaptive Exponential integrate-and-fire model (CAdEx), which solves some of the limitation of the AdEx model (Górski et al., 2021). The equations read:

$$C\frac{dV}{dt} = g_L(E_L - V) + g_L\Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) + g_A(E_A - V) + I_s \tag{31}$$
$$\tau_A\frac{dg_A}{dt} = \frac{\bar{g}_A}{1 + \exp\left(\frac{V_A - V}{\Delta_A}\right)} - g_A$$

When the membrane potential crosses a threshold, a spike is emitted, and the system is reset as in:

$$\text{if } V \geq V_D \text{ then } \begin{cases} V \to V_R \\ g_A \to g_A + \delta g_A \end{cases} \tag{32}$$

Parameters used for inhibitory (FS) populations are: $g_l = 10$ nS, $E_l = -65$ mV, $V_T = -50$ mV, $g_A = 0.$ nS, $E_A = -70$ mV, , $\delta g_A = 0$ nS, $C = 200$ pF, $\Delta_T = 0.5$ ms, $V_A = -45$ mV, $Is = 0.0$ nA, $t_refractory = 5$ ms, $V_{reset} = -65$ mV, $\tau_A = 0.01$ ms, $\Delta_A = 0.5$ mV and for the excitatory (RS): $g_l = 10$ nS, $E_l = -65$ mV, $V_T = -50$ mV, $\delta g_A = 1$ nS, $E_A = -65$ mV, $\delta g_A = 1$ nS, $C = 200$ pF, $\Delta_T = 2$ mV, $V_A = -30$ mV, $I_s = 0.0$ nA, $t_{refractory} = 5$ ms, $V_{reset} = -65$ mV, $\tau_A = 1.0$ s, $\Delta_A = 1$ mV

Then we use the Hodgkin-Huxley model (Hodgkin & Huxley, 1952b), hereafter denoted HH, with the following equations:

$$C\frac{dV}{dt} = -g_l(E_l - V) - g_K n^4(V - E_K) - g_{Na}m^3h(V - E_{Na}) + I_{syn} \tag{33}$$

with gating variables (in ms):

$$\frac{dn}{dt} = \frac{0.032(15. - V + V_T)}{(exp(\frac{15. - V + V_T}{5.}) - 1.)}(1. - n) - 0.5exp(\frac{10. - V + V_T}{40.})n$$

$$\frac{dh}{dt} = 0.128exp(\frac{17. - V + V_T}{18.})(1. - h) - \frac{4.}{1 + exp(\frac{40. - V + V_T}{5.})}h \tag{34}$$

$$\frac{dm}{dt} = \frac{0.32(13. - V + V_T)}{(exp(\frac{13. - V + V_T}{4.}) - 1.)}(1 - m) - \frac{0.28(V - V_T - 40.)}{(exp(\frac{V - V_T - 40.}{5.}) - 1.)}m$$

With $C = 200$ pF, $E_l = -65$ mV, $E_{Na} = 60$ mV, $E_K = -90$ mV, $g_l = 10$ nS, $g_{Na} = 20$ nS, $g_K = 6$ nS, $V_{Texc} = -50$ mV, $V_{Tinh} = -52$ mV.

For all types of neuron models, the parameters have been chosen in biophysical range (see (Hodgkin & Huxley, 1952b; Hille, 1992; Naud et al., 2008b; Górski et al., 2021)) in order to keep the basal asynchronous irregular activities (Brunel, 2000a) into a range of firing rate coherent with experimental observations (El Boustani, Pospischil, Rudolph-Lilith, & Destexhe, 2007; Destexhe, 2009a; Zerlaut, Chemla, Chavane, & Destexhe, 2018).

The network is built according to a sparse and random (Erdos-Renyi type) architecture where a fixed probability of connection between each neurons is set to 5% to produce pairwise Bernouilli connectivity. We consider a network model of ten thousand neurons, built according to specific properties of the cortex. This network is made of an inhibitory (FS) and an excitatory (RS) population, respectively representing 20% and 80% of the total size of the system as previouly done in (Carlu et al., 2020). The communication between neurons occurs through conductance-based synapses. The synaptic current is described by:

$$I_{syn} = g_E(E_E - V) + g_I(E_I - V) \tag{35}$$

Where $E_E = 0$ mV is the reversal potential of excitatory synapses and $E_I = -80$ mV is the reversal potential of inhibitory synapses. $g_E$ and $g_I$, are respectively the excitatory and inhibitory conductances, which increase by quantity $Q_E = 1.5$ nS and $Q_I = 5$ nS for each incoming spike. The increment of conductance is followed by an exponential decrease according to:

$$\frac{dg_{E/I}}{dt} = -\frac{g_{E/I}}{\tau_{syn}} \tag{36}$$

with $\tau_{syn} = 5$ ms

The network thus formed receives an external input, based on the activity of a third population (excitatory) of the same size as the excitatory population. Each of its neurons is connected to the rest of the network according to the same rule as mentioned earlier (fixed probability of 5 % for each connection). This external population produces spikes with a Poissonian distribution at a given tunable rate. The external perturbation that mimics the incoming seizure occurs through the augmentation of this firing rate.

The shape of the latter is described by:

$$\begin{aligned} \nu_{pert}(t) = \beta + \alpha * (exp(-(t-T_1)^2/(2.*\tau_{on}^2)) * H(-(t-T_1)) \\ +H(-(t-T_2)) * H(t-T_1) + exp(-(t-T_2)^2/(2.*\tau_{off}^2)) * H(t-T_2)) \end{aligned} \tag{37}$$

where $H$ is the Heaviside function and $\beta = 6$ Hz is the basal constant input. This function takes the general form of a high plateau, where $T_1$ and $T_2$ are the times when the perturbation reaches its beginning and end respectively, and $\alpha$ defines its maximal height. $\tau_{on}$ and $\tau_{off}$ are respectively time constants associated with the exponential rise and decay of the perturbation.

For all 3 types of networks, it is possible to have different connectivities (i.e different set of random connectivities) and realizations of Poisson drive (i.e the generator of the Poisson noise can vary). It is also possible to fix the seed of the noise either for the connectivity or for the Poisson drive (or both) to analyse specific conditions.

We create network connectivities by allowing a 5% chance of connection between any 2 neurons, which will indeed lead to an average of 5% of connection, but with some variation. Some neurons can have more afferent connections from inhibitory neurons than others, which will make them more inhibited, and the same goes with excitatory connections, creating a variation between neurons due to the random nature of the network.

## 6.2 Coarse graining and continuous analysis

In order to analyse in detail what mechanisms are at play in the network during a seizure-like event, we resort to a combination of two methods : a so-called *structural coarse-graining*, that is we gather neuron models in $n$ groups according to their inhibitory in-degree (the number of inhibitory connections they receive, as introduced before) and we study their time evolution through statistics of their membrane potential (mean and alignment) over these groups. In other words, at each integration time step, we will obtain $n$ values of mean membrane potentials, one for each group, as well as $n$ values of Kuramoto order parameter (measuring alignment in groups).

To obtain the Kuramoto order parameter, we first transform the single neurons membrane potentials into phase variables by applying a linear mapping $v_j \in [V_R, V_D] \to \theta_j^v[0, \pi]$. Then the Kuramoto order parameter is computed through the following equation:

$$R \exp i\Psi = \frac{1}{N} \sum_j \exp i\theta_j^v \tag{38}$$

$R \in [0, 1]$ gives the degree of "alignment" (if it persists in time, one would say synchronization) : $R = 1$ implies full alignment , while $R = 0$ implies no alignment whatsoever. $\Psi \in [0, \pi]$ tells us the mean phase of the transformed variables (directly related to the mean membrane potential).

Let us mention one caveat here. The membrane potentials are not mapped on the full circle, to avoid

artificial periodicity of the obtained angles: having $V = V_R$ is not the same as having $V = V_D$. One may thus ask why such a measure is used instead of the usual measures of dispersion such as the standard deviation. We use the Kuramoto order parameter because it gives a naturally normalized quantity, thus allowing a direct comparison of what is happening at each time step.

### 6.2.1 Code Accessibility

The code/software described in the paper is freely available online at `https://github.com/HumanBrainProject/PropNoProp`. The code is available as Extended Data and is run on Linux operating system.

To produce the code simulation, we used Python 3 and specifically the library brian2 (`https://brian2.readthedocs.io/`) to do the simulations of the network, add the perturbation, and save the various variables we analyzed. An example of the code to produce an AdEx network is given in the annex in 19.1.

# 7 Results

We start by showing how, in networks of various neuron models, a paroxysmal external stimulation can trigger a seizure, depending on various parameters. We show how the the dynamics can differ from model to model and what are their common features. Then, we propose a structural analysis based on the mean firing rates of individual neuron models to guide a particular coarse-graining, which we use as a filter to observe the dynamics and gain further understanding, from both qualitative and quantitative perspectives. Finally we show how this study can guide a proactive approach to reduce the chances of seizure propagation.

## 7.1 Propagative and Non-propagative scenarios

Throughout this study, we assume that the networks depicted in the previous section represent a small cortical area receiving connections from an epileptic focus. Specifically, the arrival of the seizure is modeled by a sudden rise in the firing rate of the external (afferent) Poisson region where the seizure originates. In other words, we are not concerned with how seizures *originate* (epileptogenesis), but how they *can propagate*. Therefore, we will frame our analysis into two main scenarios: *propagative*, *i.e* the network develops an excitatory firing rate greater than the input, which makes it able to propagate the seizure to efferent regions, and *non-propagative* scenario where the excitatory firing rate is lower than the input, thus attenuating the incoming signal. As described in the method section, the perturbation starts with an exponential growth followed by a plateau and ends with an exponential decrease, going back to the basal level : see blue curves in Fig.15. We show in this figure the response of the various networks to this type of perturbation.

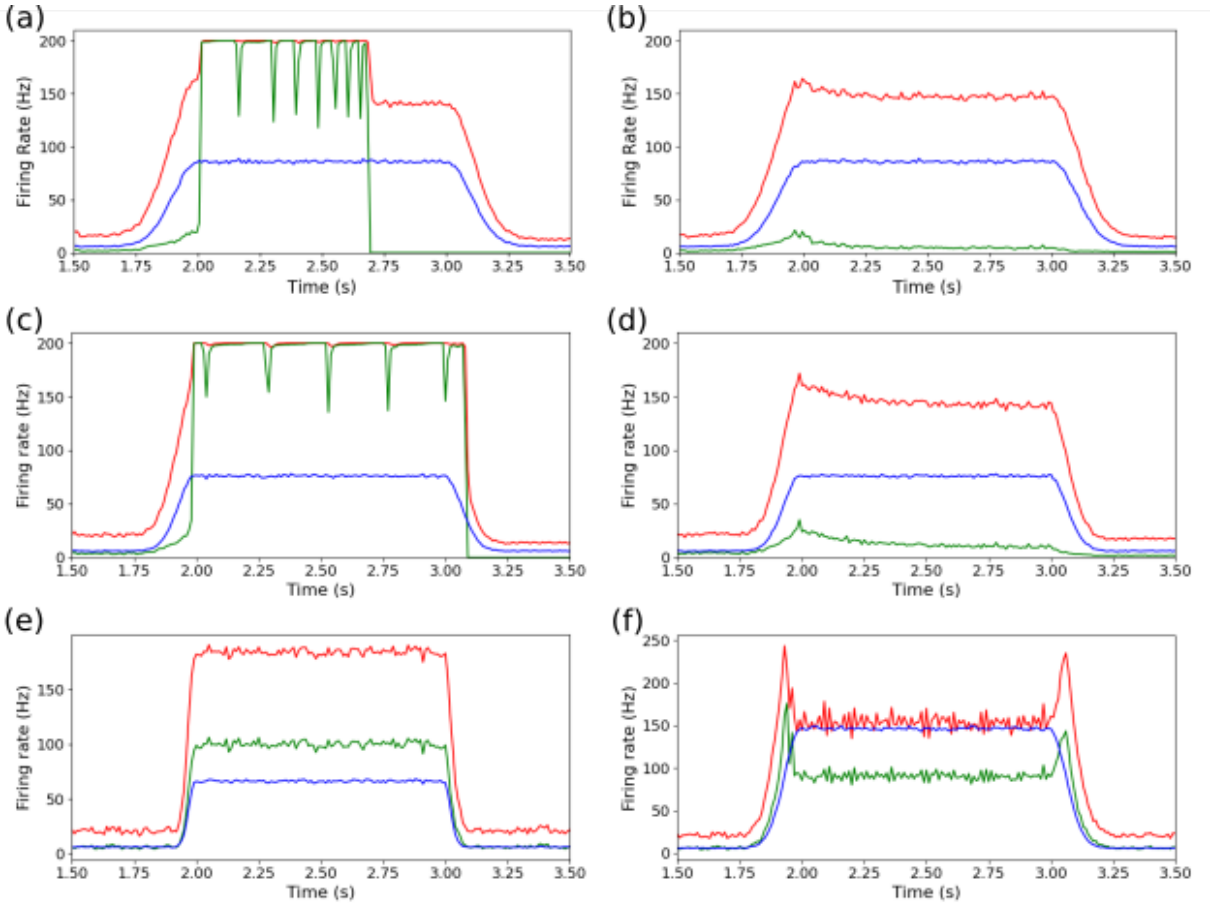


Figure 15: **a**–**f**Firing rate of the network populations in response to a perturbation (in blue the incoming perturbation, in green excitatory and in red inhibitory populations): propagative and non-propagative scenarios (respectively left and right columns) for AdEx model (**a, b**), with amplitude of perturbation $\alpha = 80Hz$ and $\tau_{on/off} = 100ms$ ; CAdEx model (**c, d**) with $\alpha = 70Hz$ and $\tau_{on/off} = 80ms$ ; HH model (**e**) with $\alpha = 60Hz$ and $\tau_{on/off} = 60ms$ and (**f**) with $\alpha = 140Hz$ and $\tau_{on/off} = 60ms$. For each model the networks are the same in the propagative of non-propagative scenarios, the only difference comes from the incoming input with different realizations.

Here we can distinguish between two classes of macroscopic differences between propagative and non-propagative scenarios.

In the first class (for AdEx and CAdEx) the difference is binary, which means the network either features a very strong increase in the firing rate of the inhibitory *and* excitatory populations, or the sharp increase in the firing rate only concerns the inhibitory population, thus strongly limiting the activity of the excitatory population (consequently preventing the seizure from spreading to the next region). From this perspective, the propagative scenario can be understood as a loss of balance between excitatory and inhibitory firing rates, which the network struggles to find once the excitatory population has reached very high firing rates. Interestingly these two scenarios can occur for the same global shape of the perturbation but changing only the noise and network realizations. It must be noted that the $200Hz$ maximum frequencies measured here are the results of the temporal binning of the global spiking dynamics, taken as $T = 5ms$, which

corresponds to the refractory time of the single neurons in Adex and CAdEx. Upon choosing a shorter binning, *e.g* $T = 1ms$, higher frequency peaks are observed, going up to $800Hz$, thus hinting at overall faster dynamics.

In the second class (HH) there is a rather continuous difference between propagative and non-propagative scenarios as can be seen in Fig.16(c), depending on the amplitude of the perturbation.

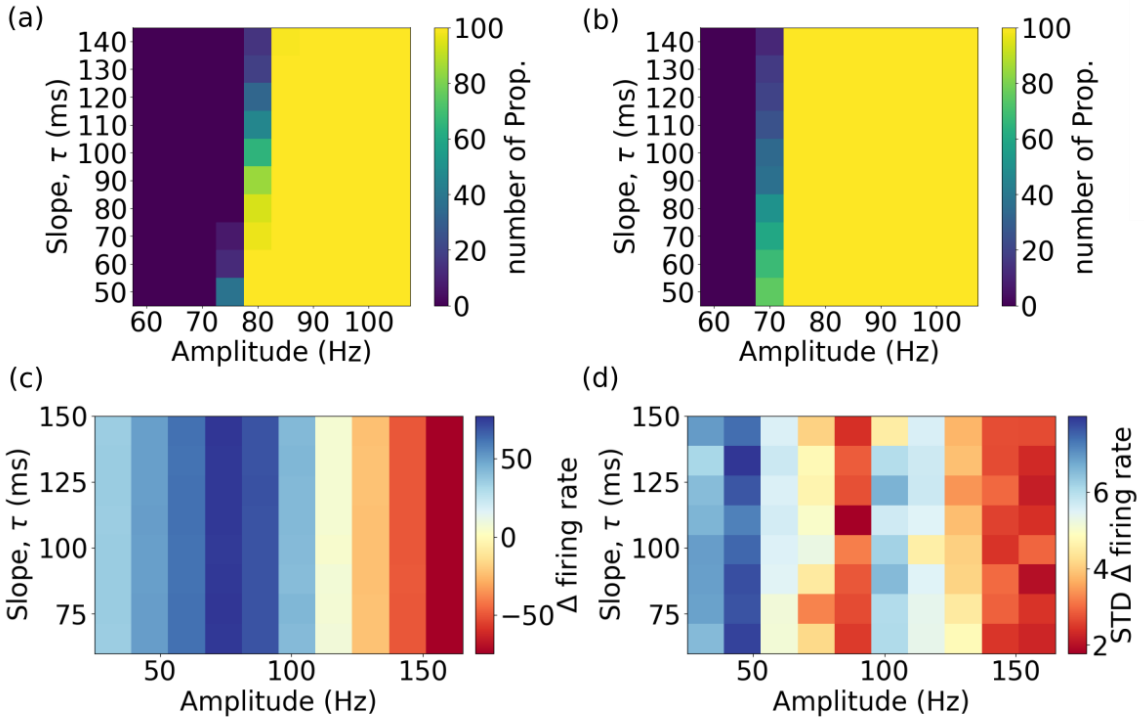


Figure 16: Grid search on the amplitude and slope of the incoming perturbation for each network. **a, b**, The percentage of realizations that propagate (Prop.), respectively, for AdEx (a) and CAdEx (b) networks. **c, d**, For HH networks, the means and SDs (over realizations) of the difference in firing rates between excitatory (**c**) and Poisson (**d**) populations ($\Delta\ firing\ rate = \nu_e - \nu_{Pois}$), averaged over the length of the plateau.

## 7.2 Influence of the perturbation's shape

To study how the shape of the perturbation affects the network's response, we screened in Fig.16 different time constants of the exponential growth rates and maximum amplitude of the plateau with 100 seeds (for both network and noise realizations for each couple of values, and probed, in the case of AdEx and CAdEx (respectively (a) and (b)), the number of realizations which yield propagative behavior, as it shows binary possible scenarios. Meanwhile, in the HH case, the perspective is a little different : we chose to show two figures, displaying means and standard deviations over realizations of the difference in firing rate between excitatory and Poisson populations (averaged over the plateau), $\Delta\ firing\ rate = \nu_e - \nu_{Pois}$ (respectively (c) and (d)). As can be expected, for all networks (AdEx, CAdEx and HH) the amplitude of

the perturbation plays a determinant role in the type of scenario we eventually find (propagative or not), however in opposite directions and of different nature. Indeed, for both AdEx and CAdEx, increasing the amplitude increases the chance of having a propagative scenario for a fixed slope, in a binary fashion, while in the case of HH the contrary is observed, and in a continuous fashion.
Also, we observe a slight coupling effect between slope and amplitude : for higher amplitudes, the propagation range extends to slower perturbations. On the contrary, in the HH network, it seems that the slope does not play any major role, hinting at a much less dynamical effect : the difference manifests itself as local equilibria of the networks under consideration, reached no matter the time course. Moreover, the standard deviations, besides showing no clear dependence on neither amplitude nor slope, are very small compared to the means, thus evidencing that noise neither plays any significant role here. These observations highlight once again the deep differences between the two types of network and their respective phenomenology.

Interestingly, in the case of AdEx and CAdEx, there exists a limit, bi-stable region here, around $80Hz$, where the perturbation may or may not propagate in the network, depending on the noise realisation. Thus, the scenario does not trivially depend on the amplitude and time constants of the perturbation in this region, which makes the latter a perfect test bed to study more deeply the internal mechanisms at play, and will thus be the main focus of the remainder of this paper.

### 7.3 Influence of structural aspects on the dynamics

In the following, we turn our attention to the bi-stable region of AdEx networks, where the two scenarios are present, and investigate what can be the source of the divergence. There are two main differences between the simulations under consideration: the realization of the network connectivity and the realization of the external input, as both rely on random number generators. We have therefore successively fixed each of them, and observed that the two behaviors were still present. Also, the global scenarios were indistinguishable from those showed so far. First, this allows us to fix the network connectivity. (which will become determinant in this part) without losing the richness of the phenomenology. Second, this tells us that what shapes the distinction between the two phenomena is more complex than a simple question of structure, or realization of the input. Another perspective is then needed to explore the internal dynamics of the network in both scenarios. As the models into consideration have very large number of dimensions, as well as quite intricate structures, brute force analytical approaches are simply not conceivable.

Let us then take a step back and investigate the relationship between the firing rate of each neuron and its number of afferent (input) connections for the three kinds of input: excitatory ($N_{inp}^{Exc}$), inhibitory ($N_{inp}^{Inh}$) and Poissonian ($N_{inp}^{Pois}$). Fig.17(a) shows the average firing rates ($\nu_E^{NP}$ and $\nu_I^{NP}$) measured over

the whole non-propagative scenario for each neuron in the AdEx network (simply defined as the total number of spikes divided by the total integration time, after having discarded a transient), plotted as a function of the three different connectivities.

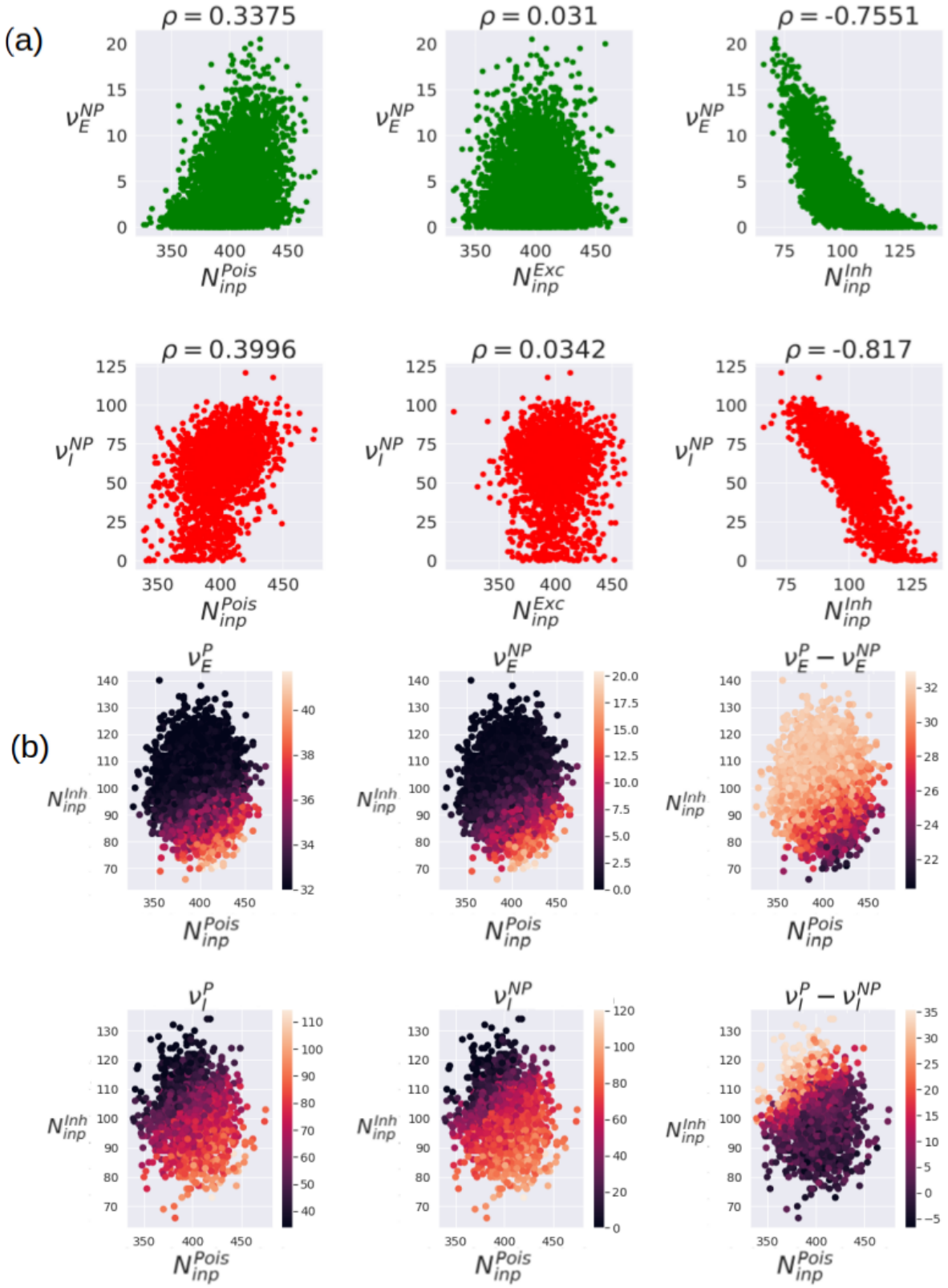


Figure 17: Influence of connectivity on single neurons firing rates: **a** Influence of poissonian ($N_{inp}^{Pois}$), excitatory ($N_{inp}^{Exc}$) and inhibitory ($N_{inp}^{Inh}$) in-degree on the firing rates of excitatory neurons ($\nu_E^{NP}$), and inhibitory ones ($\nu_I^{NP}$) in the non-propagative scenario of the AdEx network. The standard pearson correlation coefficient $\rho$ is estimated. **b** Time averaged single neuron firing rates and differences in propagative vs non-propagative regimes, as a function of both inhibitory and Poissonian in-degrees.

Note here that averaging over simulations for the sake of robustness might be a delicate matter, as we might lose constitutive differences in the process. As we are dealing with highly variable situations, we have to make compromises between generalizability and relevance. Therefore, we start with a single realization to then guide larger and more systematic investigations.

Interestingly, we see a much stronger influence coming from the inhibitory in-degree than from the Poissonian and excitatory ones. Counter-intuitively, it even seems that excitatory in-degree has almost no effect at all on total measured firing rates. Indeed, from the point of view of Pearson's correlation, inhibitory in-degree is much more (anti)-correlated with the firing rate than the excitatory in-degree (almost no correlation) or the Poissonian in-degree (little correlation). Note that we observe the same structure for propagative scenarios (results not shown). Based on these results, we can analyze whether the most salient in-degrees (inhibitory and Poissonian) has any influence on the *difference* between propagative and non-propagative scenarios, see Fig.17(b). Here, we see that the global dependency of the average single neuron firing rates on inhibitory and Poissonian connectivity does not qualitatively change between propagative and non-propagative scenarios. However, the differences $\nu^P - \nu^{NP}$ display an inverse dependency on both variables: despite maintaining a qualitative similarity between first and second columns, the seizure tends to compensate the initial disparity of firing rates. In other words, the neurons that are initially less firing, due to their structural properties, are the most impacted by the seizure. Furthermore, it must be noted that, although there is no correlation between inhibitory and Poissonian in-degrees (as can be expected from random connectivities), the third column highlights that they both play a role in the single neurons long term dynamics.

To further understand the effect of the inhibitory connectivity, we choose two points from Fig.16(a), one known to be always non-propagative, with $\tau = 70ms$ and $\alpha = 70Hz$, the other to be always propagative, with $\tau = 70ms$ and $\alpha = 95Hz$. In both situations, we varied the probability of connection from the inhibitory population between $p = 0.04$ and $p = 0.06$, as shown in Fig.18. Note that the influence of the incoming inhibitory connectivity shifts the boundary between propagative and non-propagative behaviors. This is an important influencing factor in relation to the shape of the perturbation and in particular its amplitude.

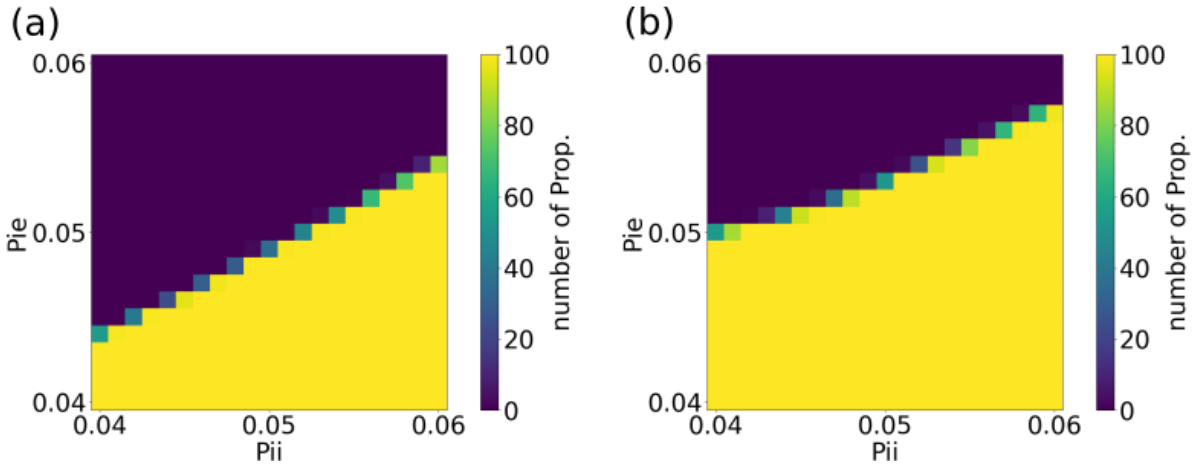


Figure 18: Grid search on the in-degree inhibitory probability of connection for the AdEx network.**a, b**, Percentage of propagation (Prop.) with parameters, as follows: $\alpha = 70Hz$ and $\tau = 70ms$ (**a**) and $\alpha = 95Hz$ and $\tau = 70ms$ (**b**) , where for both figures Pie is the probability of connection from inhibitory to excitatory neurons and Pii is the probability of connection from inhibitory neurons to inhibitory neurons. Decreasing the probability of connection from inhibitory to excitatory neurons or increasing the probability of connection from inhibitory to inhibitory neurons, tends to decrease the overall inhibition in the network and thus facilitates propagative behavior.

It is worth pointing out that these results establish a clear link between structure and dynamics, but structure is by itself not a sufficient criterion to understand the underlying mechanisms. We therefore focus on the temporal evolution of the propagating activity.

Beforehand, we take a step back and probe whether the differences in the individual mean firing rates are associated with specific roles in the dynamics. To achieve so, we start classifying, for the AdEx network, the neurons' indices in the raster plot according to the total number of spikes they emit during the whole simulation. We chose for this purpose a representative propagative scenario.

The sequence of propagation of the perturbation then appears visually in Fig.19(a). We observe, in the case of propagation, a fast cascade (consistent with the experimental observations (Neumann et al., 2017)) : some neurons are quickly driven into a sequence at the onset of the seizure.

In addition, there is no perfect synchronization of the action potentials of all neurons. This is an interesting result, coherent with experimental observations on epilepsy (Jiruska et al., 2013).

Secondly, we examine the same situation, but sorting neuron indices as a function of the number of inhibitory inputs they receive, as shown Fig.19(b), as it is the most influential structural feature we observed in our model. Here too, the cascade phenomenon is clearly visible, indicating that the inhibitory input connectivity has a central influence on the dynamics at play during the perturbation in the propagative scenario.

Fig.20 shows the same pictures for CAdEx and HH networks. We see here that CAdEx network's behaviors are very similar to AdEx : sorting with firing rate or inhibitory in-degree gives very similar

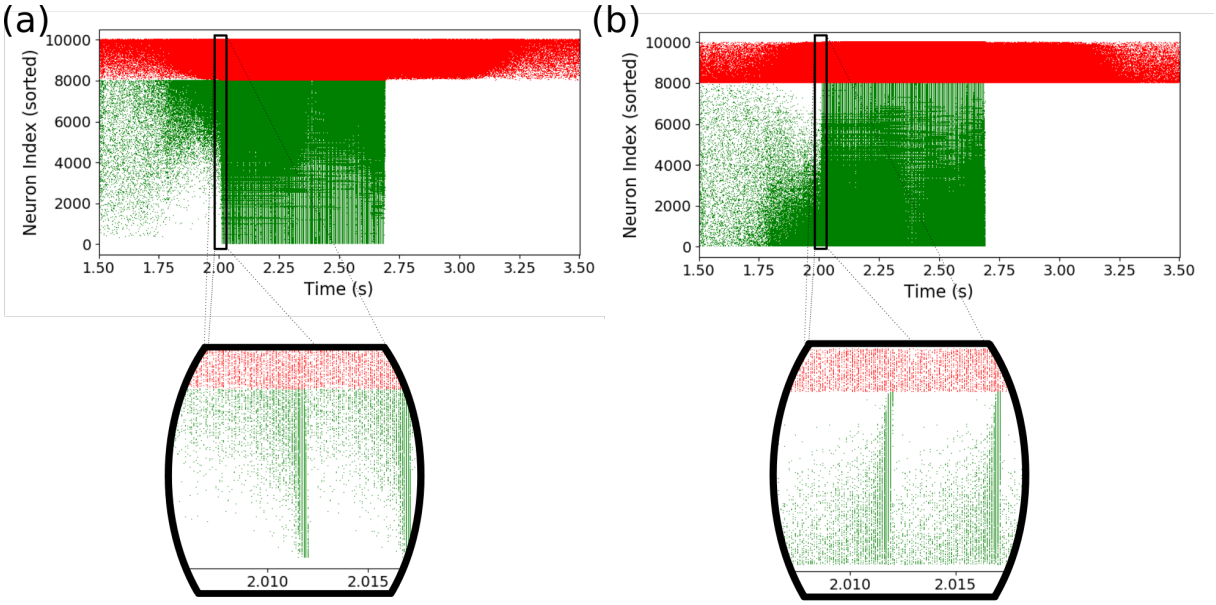


Figure 19: Dynamics in the propagative scenario (AdEx). **a**, In a raster plot of a simulation with propagative behavior, neuron indices are sorted according to the number of spikes during the simulation. A "cascade" phenomenon can be observed when zooming on the onset of the perturbation propagation in the excitatory population. **b**, The same cascade phenomenon is observed when neuron indices are sorted as a function of the number of inhibitory inputs received. Note that the absence of excitatory activity after the perturbation is because of a strong adaptation current (Eq.(29), and Eq.(30)).)

structures and we can distinguish here too the cascading effect at the onset of the perturbation, following the indices. HH networks show quite a different phenomenology. First the two sorting do not show the same structures, which hints at a more subtle mapping between inhibitory in-degree and long-term single neuron model dynamics. In the firing rate sorting, we can still distinguish 3 blocs of distinct activity, and thus of populations, corresponding to the 3 key periods of the simulation : before stimulation, at the onset, and during the stimulation. Interestingly, it seems that before and during the stimulation different populations of neuron models are distinctly mobilized. While before the stimulation, the central neurons (with respect to their indices) are active, a double cascade contaminates the rest of them (towards higher and lower indices) at the onset, ending in a general surge of activity. This must be contrasted with the in-degree sorting panel, where the cascade is more unidirectional, as the main activity slides from low connectivity indices (less connected) to the higher ones, until all neurons fire. This emphasizes the importance of the perspective chosen to analyse complex behavior : none of these perspectives alone completely explains the intricate interplay between structure, long term, and short term dynamics.

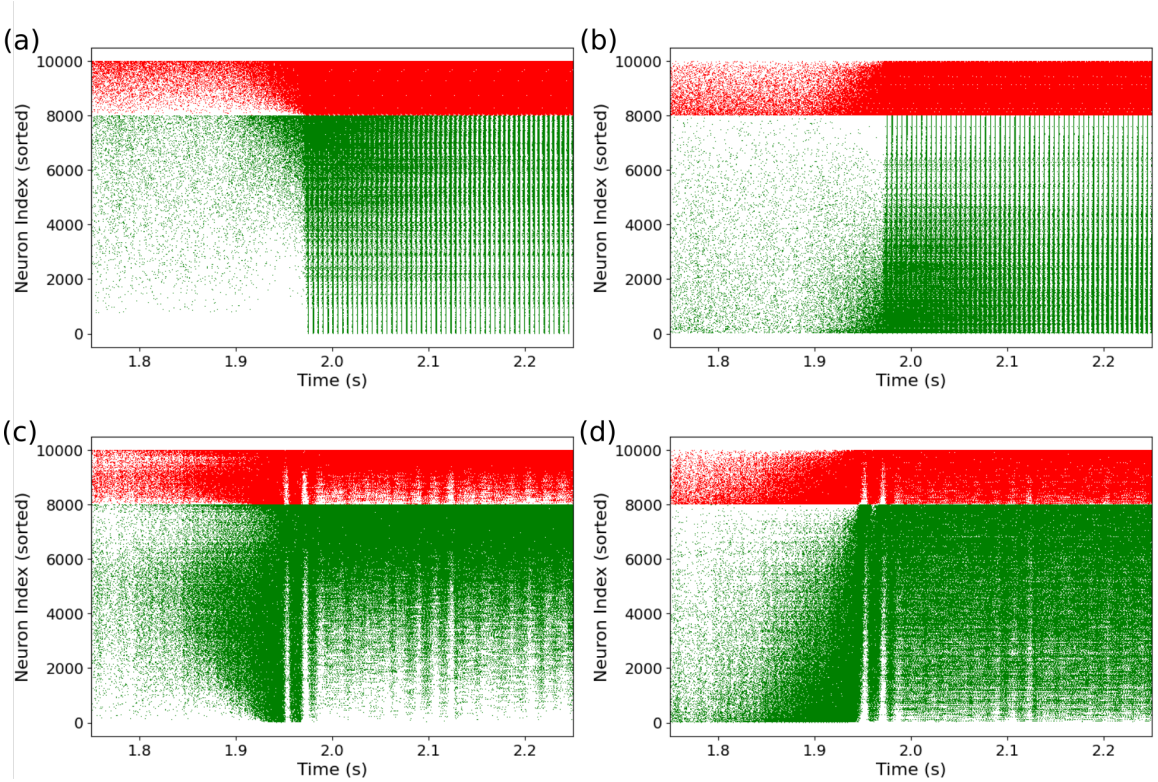


Figure 20: Dynamics in the propagative scenario (CAdEx and HH). **a**–**d**, Same plot as previously shown but for the CAdEx network (spike sorting, **a**; inhibitory in-degree sorting, **b**) and HH network (spike sorting, **c**; inhibitory in-degree sorting, **d**). Cascade phenomena are still observable in **a**,**b** and **d** hence showing its robustness, but not in **c**, where propagation takes a slightly different form, highlighting the contrast induced by different perspectives on a single-complex dynamics.

Altogether these results show the relevance of adopting a perspective based on the inhibitory in-degree : it gives an operational method to rank single neurons, and this ranking is clearly associated with specific dynamical features, hence allowing us to study the role of the internal organisation of the network before and during the paroxysmal event. As the cascade phenomenon is similarly visible in all types of networks, in the next section we focus on the AdEx network. We push further this analysis by comparing propagative and non-propagative scenarios, and make use of the continuous measures introduced in Material and Methods.

## 7.4 Continuous measures on subgroups of neurons

Focusing on the AdEx network, we first consider groups of neurons defined by their inhibitory in-degree. Note that these are somewhat artificial, as they are only statistical reflections of topological aspects of the network (i.e, there is no reason to think a priori that all neurons having $n$ inhibitory inputs would have more privileged links among themselves than with those having a different number. However, they allow in principle a variable degree of categorization, based upon the sampling of the inhibitory in-degree distribution, which eventually leads to different levels of (nonlinear) coarse-graining (although we will consider only one such sampling here). Secondly, we switch our analysis to continuous variables, which

allow a finer and more systematic analysis of the dynamics, as they don't depend on spike times. Indeed, although spike timings are the most accessible collective measures in real-life systems, which make them the most fitted candidates for "transferable" studies, we want here to take advantage of the virtues of mathematical modeling to probe the underlying mechanisms in these simulations, to then be able to draw conclusions on more accessible observables. We focus here uniquely on membrane potentials, as they are the closest proxy of the firing dynamics in the network and chose to use two main measures based on them: the mean $\mu_V$ and a modified Kuramoto order parameter $R$, which gives a naturally normalized measure of instantaneous alignment (or similarity) of the membrane potentials. Both are defined in time, over a class of neurons. As randomness plays a crucial role in our simulations, through network connectivity as well as noise realization, it is important to control how much it affects the results we obtain. To achieve so, we start by fixing the network connnectivity while averaging over noise realizations, and then average over connectivities while looking for noise realizations that lead to propagative and non-propagative scenarios for each structure.

**Mean membrane potential in time**

In Fig.21(a)-(b), the mean membrane potential $\mu_V$ defined for each group of excitatory (RS) and inhibitory (FS) neurons, in time. The top and bottom rows respectively refer to the averages and standard deviations over noise realizations (input), as the network connectivity is held fixed. For propagative (Fig.21a) and non-propagative (Fig.21b) scenarios, all the data presented from now were obtained by regrouping neurons having the exact same inhibitory in-degree, thus corresponding to a discrete one-to-one sampling of the input distribution. Note that, given the network architecture under consideration, the number of afferent inhibitory synapses defined over both populations of neurons follows a binomial distribution with a mean around 100 connections. From that, we arrange neurons in groups of identical number of inhibitory connections, which gives us about 60 groups (varying with population and connectivities) containing at least 1 neuron.

To confirm that our results were not depending on the specific connectivity we had we simulated 50 different networks with different connectivities (otherwise being identical) and found a couple of noise realizations for each corresponding to propagative and non-propagative scenarios. Those various networks still have the same meta-structure and follow similar statistics. They only show that within those specific choices, the variations that exist do not impact the results we show. We applied the same method to create the different groups, but the number of said groups could differ due to the random variability in the connections. Therefore, many"extreme" groups are poorly represented among the various connectivities, which would make them hard to average over. We thus discarded them. The average and standard deviation of $\mu_V$ over the 50 different connectivities is shown in Fig.22(a)-(b). The white lines could be

a weak manifestation of the previous effect, which made the standard deviation very high, coupled with the fixed range of color scales, imposed for the need of clarity. We observe that this figure looks very similar to Fig.21, which suggest that the results are not limited to a specific connectivity.

We see from Fig.21 and Fig.22 that the inhibitory in-degree profile seems to play a major role in the overall dynamics. Indeed, as the perturbation is growing (starting $250ms$ before the maximum at $2s$), we can first observe a strong increase of the mean membrane potential of all excitatory neurons, starting from low indices, then followed by a low-potential cascade, also starting from low indices and then contaminating to higher ones.

This latter effect is much clearer in the case of inhibitory neurons, where the cascade follows very well the input profile, in both propagative and non-propagative scenarios. Note that the low-potential area can be easily understood as a high-firing regime: neurons fire as soon as they leave their resting potential, thus displaying very low values of membrane potential when calculated (and sampled) over time.

Interestingly these pictures show that, up to the decisive point of the seizures, the continuous measures look very similar, thus hinting at an instantaneous finite-size fluctuation causing the whole network to explode. Also, it is noteworthy that the new "hierarchy" set by the cascade is conserved in the non-propagative regime, while the propagative regime seems to have an overall reset effect.

Also, we see from these graphs that there is a particular time window where the variance of the mean membrane potential is larger for the most inhibitory-connected neurons, in both RS and FS populations (although it appears clearer for RS ones here, because of the need to rescale the FS colorbar to have comparable results). This increase of variance, while still present, is weaker and on a smaller time window in the average over connectivities compared to the average over noise realizations. This suggest connectivity plays a role in the intensity of the effect, although it remains qualitatively similar. We found that this time window defines the period when the network can actually switch to propagation: the high variance corresponds to different times when various realizations "explode", and thus defines a region of instability.

A central point to raise here is that what makes the difference between propagative and non-propagative scenarios is most likely *not* an *infinitesimal* instability defined from a macroscopic perspective [i.e., that is because of a positive eigenvalue of a Jacobian defined from a large-scale representation (e.g., mean-field)], otherwise the non-propagative behavior would simply not be observable (as, except for chaotic dynamics, we do not observe unstable trajectories in phase space). Indeed, what differs between the various simulations is either the noise, or the connectivity realization, which may, or may not, bring the system to a point of instability. The external Poissonian drive, with *finite-size* fluctuations is thus constitutive of the scenarios we observe.

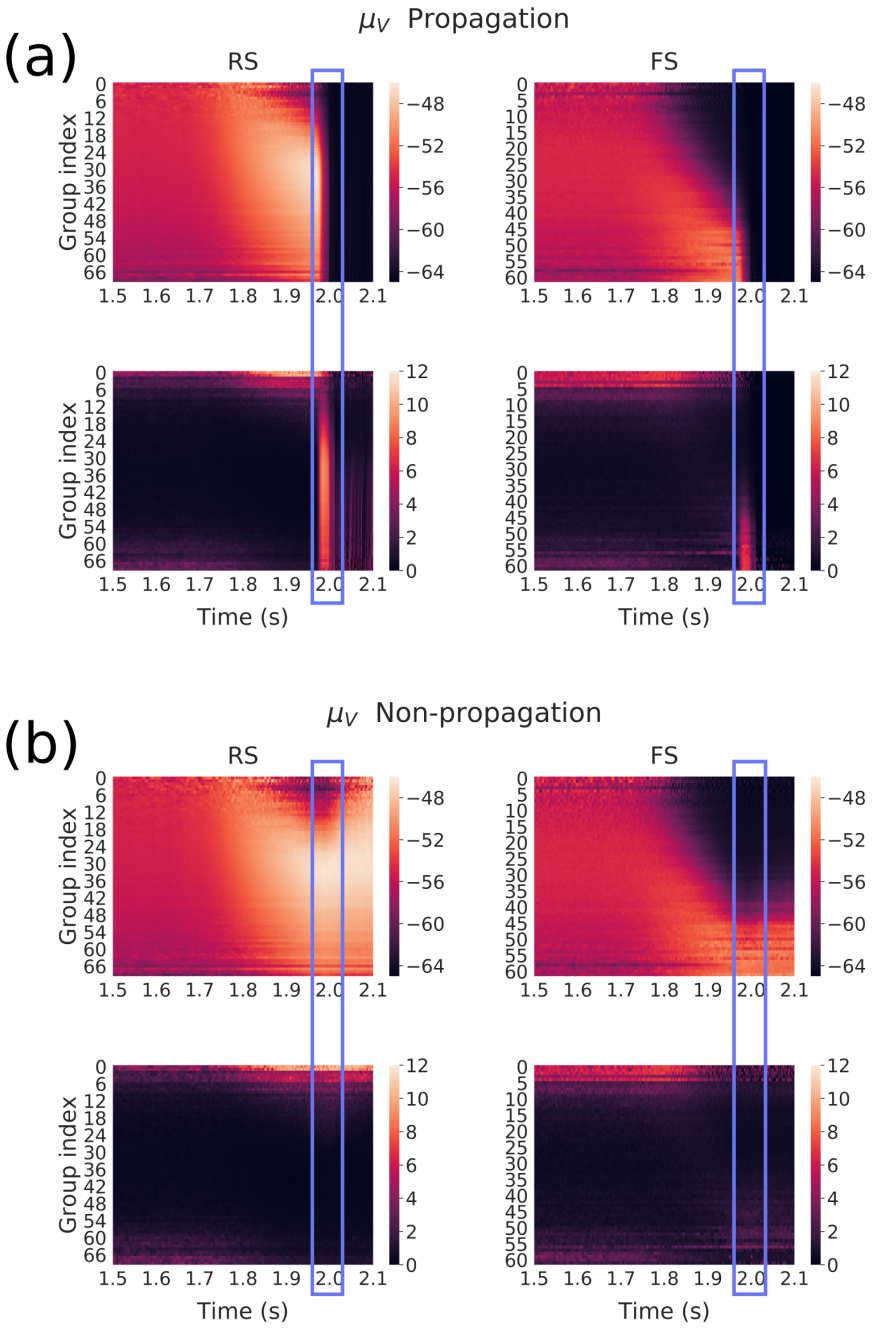


Figure 21: Mean membrane potential over subgroups of neurons (same network connectivity, different noise realizations) for each group defined as a function of their incoming inhibitory connections (from top to bottom : least amount of inhibitory connectivity to most amount), averaged over 50 noise realizations (17 non-propagative and 33 propagative). **a, b**, Color maps correspond for each group to the average membrane potential (top) and SD (bottom) across noise realizations in the propagative situations (**a**) and non-propagative situations (**b**) for both excitatory (RS) and inhibitory (FS) populations. The blue rectangle highlights the (time) region where the system either switches to a propagative regime or remains stable.

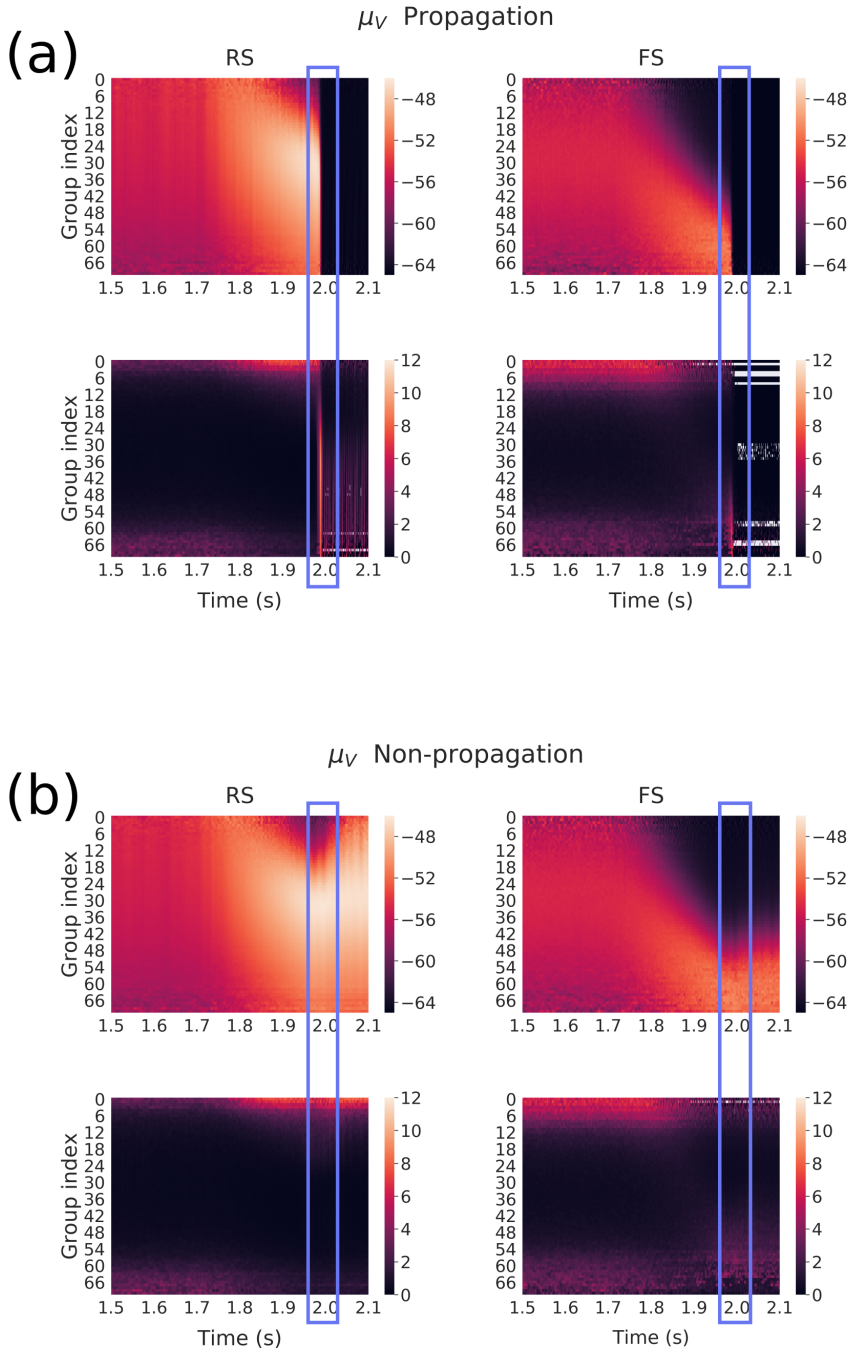


Figure 22: Mean membrane potential over subgroups of neurons (different network connectivities) for each group defined as a function of their incoming inhibitory connections (from top to bottom : least amount of inhibitory connectivity to most amount). Here, we averaged over 50 network connectivities, for which we found a couple of noise realizations corresponding to propagative and non-propagative scenarios. **a, b**, Color maps correspond for each group to the average membrane potential (top) and SD (bottom) across different connectivities in the propagative (**a**) and non-propagative scenarios (**b**) for both excitatory (RS) and inhibitory (FS) populations. The blue rectangle highlight the (time) region where the system either switches to a propagative regime or remains stable.

To gain more insight into the diversity of dynamics across neuron groups, we turn our attention to a measure of alignment, or synchronisation, namely the Kuramoto order parameter $R$.

**Kuramoto order parameter**

The Kuramoto parameter represents a degree of alignment, a value of 0 meaning there is no alignment while a value of 1 meaning everyone is perfectly alined. We show in Fig.23(a)-(b) the Kuramoto order parameter $R$ defined for each group of excitatory (RS) and inhibitory (FS) neurons in time, averaged over noise realizations (top row), and standard deviation over realizations (bottom row), in propagative (a) and non-propagative (b) scenarios (network connectivity held fixed).

Again, we reproduce the results with 50 network connectivities, for both propagative and non-propagative scenarios, see Fig.24(a)-(b). We clearly see that the results are qualitatively similar, although with seemingly higher contrast than Fig.23.

The cascade previously observed is clearly visible for the average $R$, in the form of a "desynchronization cascade". For the propagative scenario, we note here a *recruitment* process between two radically different regimes having nonetheless alignment features: a fluctuation-driven asynchronous irregular (AI) dynamics, where membrane potentials are mostly conditioned by the balance of inhibitory versus excitatory inputs, and a seizure characterized by high spiking and membrane potentials clamped by refractoriness. Interestingly, in the non-propagative scenario, it appears that the misalignment of the inhibitory neuron groups finally attained is fueled by the joint activity (of the network and the input), thus hinting at a out-of-equilibrium steady state (that continues until the end of the plateau of the perturbation, $1s$ later). From the standard deviation perspective, wo main features are worth pointing out. First, we again observe the instability window, characterized by high standard deviation between realizations in propagative scenarios. Secondly, we see that the two types of averaging leads to strikingly similar results, although slightly different quantitatively speaking, the average over connectivities leading to a higher contrast during the cascade. Therefore, our results are independent of both the noise realization and the specific connectivity, although an average over one or the other is useful to observe a typical case.

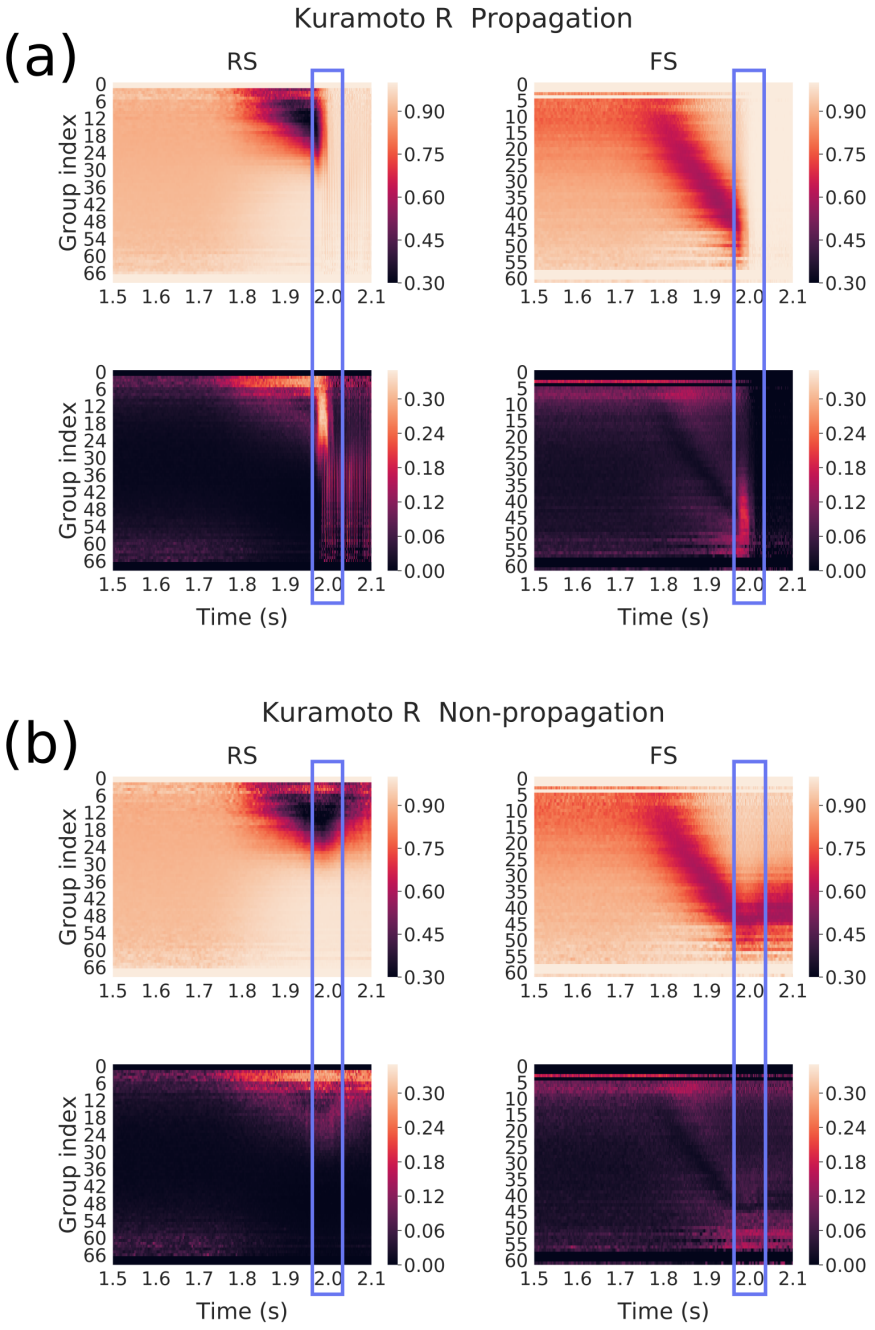


Figure 23: Kuramoto R of membrane potentials over subgroups of neurons (same network connectivity, different noise realizations) for each group defined as a function of their incoming inhibitory connections (from top to bottom : least amount of inhibitory connectivity to most amount), averaged over 50 noise realizations (17 non-propagative and 33 propagative). **a, b**, Color maps correspond for each group to the average Kuramoto parameter (top) and SD (bottom) across noise realizations in the propagative (**a**) and non-propagative (**b**) scenarios for both excitatory (RS) and inhibitory (FS) populations. The blue rectangle highlights the (time) region where the system either switches to a propagative regime or remains stable.

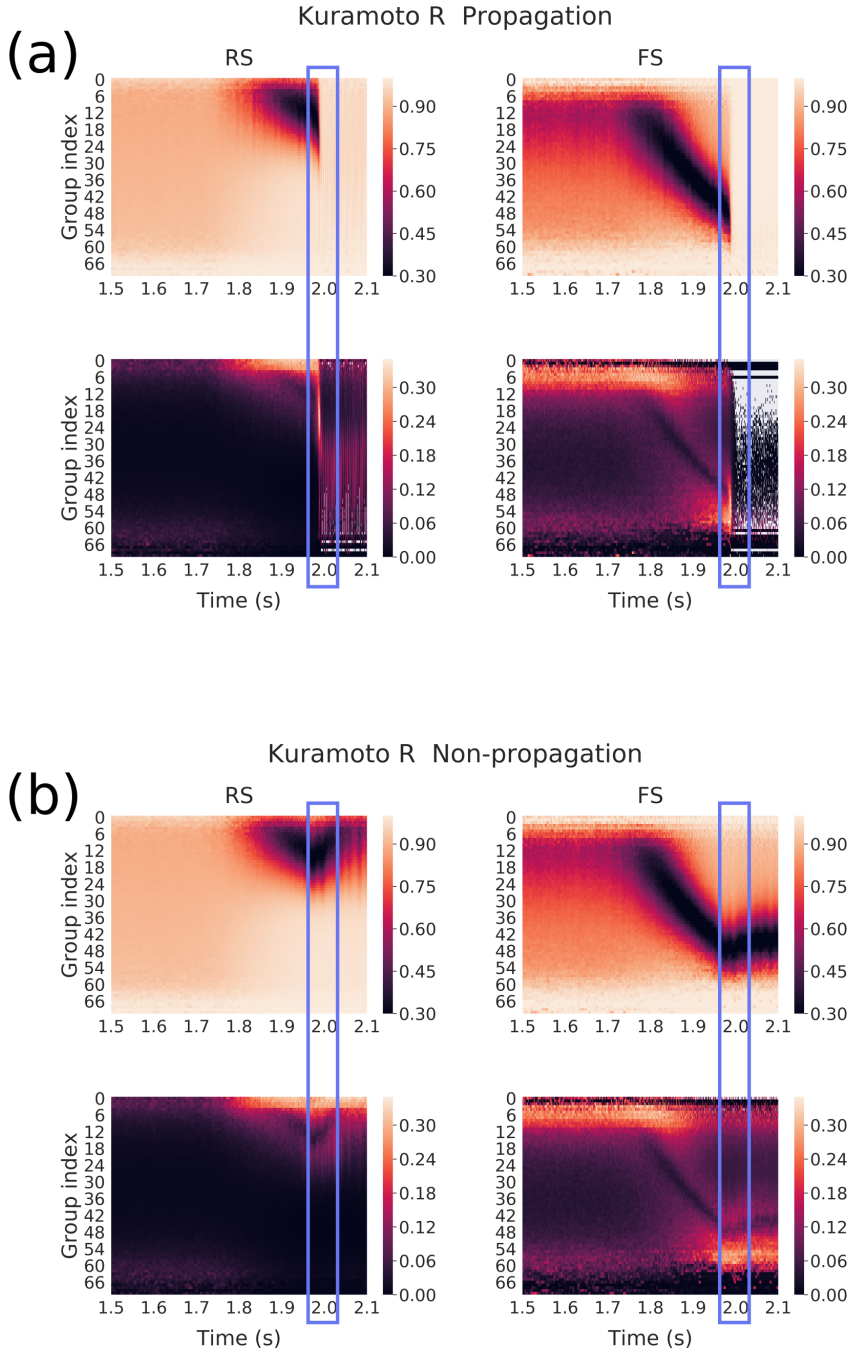


Figure 24: Kuramoto R of membrane potentials over subgroups of neurons (different connectivities) for each group defined as a function of their incoming inhibitory connections (from top to bottom : least amount of inhibitory connectivity to most amount). Here, we averaged over 50 network connectivities for which we found a couple of noise realizations corresponding to propagative and non-propagative scenarios. **a, b**, Color maps correspond for each group to the average membrane potential (top) and SD (bottom) across different connectivities in the propagative ( **a**) and non-propagative scenarios ( **b**) for both excitatory (RS) and inhibitory (FS) populations. The blue rectangle highlights the (time) region where the system either switches to a propagative regime or remains stable.

## 7.5 Dynamic versus static approach

We have seen that changing the slope and the amplitude of the signal alters the chances of triggering a seizure, thus hinting that the time evolution of the perturbation is central. Then we observed a hierarchical structure setting in from the point of view of continuous measures, following the perturbation. However, fundamental questions remain: how much of this latter phenomenon is actually dynamic? Would we find the same structures if we bombarded the network with a fixed input at, say, $80Hz$? Can we observe the same dynamical structures for scenarios which are always, or never, propagative (no matter the noise or connectivity realization) ? This would indicate that the structures observed thus far might have little to do with the seizure phenomenology itself but would either be the mere results of strong conditioning of the network by the level of input (if static structures are similar), or simply not yield any explanation for the instability we observe (if always/never propagative scenarios show similar features).

We now turn our attention to Fig.25(a), which displays the static $\mu_V$ profiles in RS population obtained for fixed external inputs ("Stat." curves), together with the profiles captured at the typical onset of the seizure, for various amplitudes: 60Hz (never propagative), 80Hz (sometimes propagative) and 100Hz (always propagative). The network realization is the same as previously analyzed, except when explicitly stated (Net. 2), where we refer to another connectivity realization. For the 80Hz scenarios with the first network (the one we have been investigating so far), we kept the splitting of the realizations between propagative and non-propagative, to highlight the potential differences of structures.

First, as previously observed, the profiles obtained for propagative versus non-propagative regimes are very similar for lower values of inhibitory connectivity. Then, we clearly see that the $\mu_V$ profiles extracted from the dynamical situations (hereafter called the dynamical profiles) are very different from the static ones.

Besides, it is worth pointing out that the profile obtained for a 80Hz amplitude with a different random realization of the network (where all 50 noise realizations are put together, based on the previous observation that propagative and non-propagative scenarios show very similar structures) is very similar to those already shown, with small standard error, which, together with the previous observation that noise and network realizations seem to play similar roles, underlie a robust network phenomenology. Furthermore, we see that the profiles obtained for 60Hz, 80Hz and 100Hz amplitudes *are different*. The nature of their differences is of great interest for low indices, where we observe that 60Hz and 100Hz profiles are located on opposite sides of the central 80Hz profile: their ordering in this region is consistent with that of their response to the perturbation we have observed so far (see Fig.16). This said, the dynamical profiles yet show similar qualitative features : they all are non-monotonous and display two well-separated parts. Indeed, for low indices (until 30) $\mu_V$ is increasing with values starting around the lowest of the static profiles ( 10Hz), while their high indices part is more aligned with high static profiles. Interestingly,

we see that for 60Hz and 80Hz the right part is well aligned with the static profile obtained for similar inputs. This does not seem to be the case for 100Hz, although the static input simulation displays some instability, which makes their comparison less relevant. Although it is not straightforward to link $\mu_V$ with the instantaneous regime, we have seen that low values can be associated with high firings (the neurons spending most of their time clamped at $-65$mV). This helps understanding what is happening here: for higher values of amplitude, the less inhibitory-connected neurons are firing more, and can thus lead the rest of the network to higher activities.

Fig.25(b) shows the Kuramoto order parameter aspect of the latter figure. Here the $R$ profiles display structures quite different from those observed for $\mu_V$. Indeed, the various static profiles do not display such clear variability as for $\mu_V$, although little differences can still be observed: high inputs seem to show more variability in low indices, while ending at higher values for higher indices. More importantly, the dynamical profiles are here very different, from the static ones, and among themselves. Besides, the simulated propagative and non-propagative scenarios show little differences here as well, and the profiles corresponding to same amplitude (80Hz) and different network architecture (Net. 2) also overlap here. Interestingly we can also observe that the 60Hz and 100Hz profiles are different and located apart from the 80Hz, although they also show different magnitudes of their inverted peaks. Given that the ordering of these magnitudes are not consistent with the various degrees of instability, we suggest that the *position* of the peak might be the most discriminating factor to establish whether the scenario is propagative. This would be consistent with the observations we made thus far, and confirm our previously suggested scenario: as the more we approach the center group, the more neurons are considered (Binomial distribution), the green peak (100Hz) tells us that more neurons have undergone the desynchronization cascade we mentioned earlier, that is, more neurons have already “switched side" and entered a high firing regime, thus giving more inertia to the cascade phenomenon. The middle scenario (80Hz) would then sit on a *tipping point*, that is a point separating two radically different dynamical regimes of the system.

These latter observations show that, from the perspective of both mean membrane potential and Kuramoto order parameter calculated inside the groups formed from inhibitory in-degree, we are in the presence of a structured behavior which emerges from an intricate interaction between dynamics and architecture, and which cannot be recovered from static approaches.

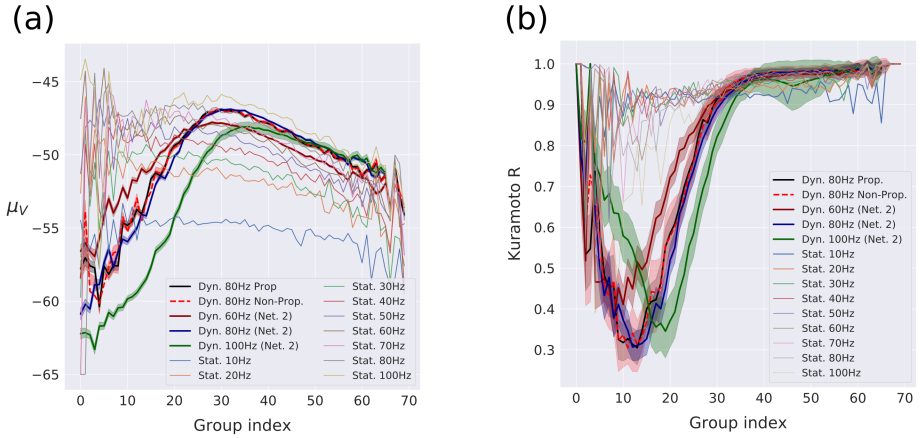


Figure 25: **a,b**Steady-state and dynamical profiles of RS neurons for (**a**) $\mu_V$ and (**b**) Kuramoto $R$ over subgroups of neurons [same network connectivity (unless specified), different noise realizations], for fixed external input. The steady-states, (Stat), represent the stable activity without perturbation. They are drawn together with various profiles for different amplitudes of perturbation (Dyn) captured right before seizure onset, at respectively 1950ms (60Hz), 1950ms (80Hz), 1930ms (100Hz, as the seizures develop *before* 1950ms). Networks are the same as previously analyzed, except when stated Net. 2 which represent another network connectivity, for robustness. Standard errors estimated over noise realizations are shown in shaded areas.

## 7.6 Can seizure propagation be controlled by external inputs?

After having established that the structure of the dynamics allows or not the propagation of the paroxysmal perturbation, we now investigate whether we could use the previous finding of a strong instability window for the 80Hz dynamical scenario to alter the fate of the AdEx network dynamics. This approach is based on the following reasoning : we have observed, with a detailed analysis, that switching to one scenario or another is determined in a short time windows (just before the eventual seizure). Thus, we want to design a stimulation protocol to reduce the chance of seizure propagation, based on this observation, *but which does not require the same level of analysis*, hence making it applicable inline and without the need of extensive computational power. To do so, we will study the region around the seizure to determinate this relevant time window.

To achieve so, we apply a Gaussian stimulation, with 10 ms time constant, two different amplitudes (1Hz and 5Hz), positive or negative, *through a variation* of the external excitatory input (which depending on when the simulation is applied, can be the drive of 6Hz or the drive plus a value between 0 and 80Hz, depending on the timing as we have a perturbation of the shape of 80Hz with a time constant of 100ms). For simulations performed under the same conditions, the stimulations were applied at different times as detailed in Tables 1(a)-(b). These tables show, for a total number of 100 simulations (with same network Connectivity but different noise realizations), among which 72 were propagative, what relative percentage of simulations has undergone a triggering and a cancellation of the seizure, respectively. In this tables, the perturbation reach its maximal value of 80Hz at $2000ms$.

(a)

| time of peak | +1Hz | +5Hz | -1Hz | -5Hz |
|---|---|---|---|---|
| t = 1500ms | 0.1806 | 0.1944 | 0.1528 | 0.1389 |
| t = 1850ms | 0.1389 | 0.1944 | 0.0972 | 0.1528 |
| t = 1950ms | 0.1528 | 0.2361 | 0.125 | 0.0694 |
| t = 1975ms | 0.0972 | 0.0 | 0.3472 | 0.3889 |
| t = 2000ms | 0.0139 | 0.0 | 0.25 | 0.5556 |
| t = 2500ms | 0.0 | 0.0 | 0.0 | 0.0972 |

(b)

| time of peak | +1Hz | +5Hz | -1Hz | -5Hz |
|---|---|---|---|---|
| t = 1500ms | 0.25 | 0.1786 | 0.2857 | 0.25 |
| t = 1850ms | 0.178 | 0.1786 | 0.2143 | 0.2143 |
| t = 1950ms | 0.0357 | 0.6071 | 0.2143 | 0.5 |
| t = 1975ms | 0.7143 | 1.0 | 0.25 | 0.28572 |
| t = 2000ms | 0.6071 | 1.0 | 0.0 | 0.0714 |
| t = 2500ms | 0.0 | 0.0 | 0.0 | 0.0 |

Table 1: **Triggered and prevented events:** (a) Percentage of prevented events, from 72 initially propagative behaviors. Highlighted in orange $\geq 25\%$ and in red $\geq 50\%$. The time of peak corresponds to the moment where the maximum of the stimulus is reached, the amplitude corresponds to a variation of the external input (see the main text) (b) Percentage of triggered propagation events, from an initial number of 38 non-propagative cases. Highlight in orange $\geq 25\%$ and in red $\geq 50\%$.

We see that it is possible to “reverse" the scenario from propagative to non-propagative in the time windows between 1975 ms and 2000 ms (and to vice versa, albeit for a larger time window) thanks to (or because of) the stimulation as can be seen in (Table 1(a)) (see Table 1(b) for the opposite).

A notably interesting case is that more than 50% of the seizures are prevented if a stimulation of -5 Hz is applied in the same time window. This could open interesting leads in furthering qualitative comparisons between computational simulations and real-life situations, and eventually guide future interventions.

# 8 Discussion

In this computational work, we studied the response of various spiking neural networks to paroxysmal inputs. We observed that the same networks can display various types of responses, depending on its nature (the neuron model used at its nodes), the shape of the perturbation (here we analysed particularly a plateau-like input with various slopes and amplitudes) and the realization of the random number generator. In the case of AdEx and CAdEx networks, two radically different responses to a qualitatively similar incoming excitatory perturbation are observed. Indeed, the latter could either recruit the excitatory population and thus allow the seizure to propagate to efferent areas, or be “controlled” by the activity of the inhibitory population, keeping the excitatory population at a low activity level, thus preventing further propagation. The response of the network depends not only on the amplitude of the perturbation but also on its rising speed. This is consistent with experimental observations (Saggio et al., 2020). Interestingly, in the case of a HH network, our investigations show very different network responses, where mostly the amplitude of the perturbation plays a role and where no variability on noise realizations was observed.

A rich literature shows that seizures can be classified according to their onset/offset features described by bifurcation types(Saggio et al., 2020; Saggio, Spiegler, Bernard, & Jirsa, 2017; Jirsa, Stacey, Quilichini, Ivanov, & Bernard, 2014). The most observed bifurcation at the onset of a seizure is a saddle-node bifurcation (Saggio et al., 2017), which is characterized by an abrupt change in the baseline of the electrophysiological signal (Jirsa et al., 2014). We observed in the current work that seizures are propagative in AdEx and CAdEx networks when they rise abruptly enough in the network. There is here an interesting correspondence revealing the importance of the onset of seizure dynamics, as it has been shown from a clinical point of view (Lagarde et al., 2018). It is worth noting that the absence of such phenomenology in HH networks (for the scenarios we considered) raises interesting questions in the modeling of seizure dynamics, but also more generally in neuronal networks : how the quantitative differences (number of variables) and qualitative differences (types of processes taken into account) in the single neuron models affect the global dynamics ? Are more precise models always the best in all respects ? This underlines the importance of the choice of model and of parameters: by modeling a neuronal network and observing a phenomenon which resembles reality, we are not testing whether the specific ingredients we chose *are* constitutive of this phenomenon, but *how they would be* if they were chosen a priori. It is only the systematic cross-model observations and comparisons that can yield such an answer as which are the necessary and sufficient ingredients to observe a given phenomenon.

Note that, in clinical observations, the most accessible measurements are made on a macroscopic scale. In the study proposed here, we observe the activities at a smaller descriptive scale by building a network of neuron models. We thus have a complex system of very high dimension, rendering a priori impossible to obtain a simple description of the dynamics, which motivates the statistical approach proposed here. With this type of analysis, we were able to track in time key features of the underlying dynamics, especially those supported by the structure of the network : inhibitory in-degree can be mobilized to explain global differences in network response. Indeed, we proposed a coarse-grained description of the network dynamics based on inhibitory in-degree, allowing us to capture internal processes that were not visible at first, and which play a significant role in the global out-of-equilibrium dynamics. We chose inhibitory in-degree as it was found to be the most influential aspect determining the firing rate (see Fig.17(a)). It is interesting to note inhibitory neurons were also the ones that had the highest firing rate (around $15Hz$) while the excitatory neurons were way lower (around $2Hz$) and the Poisson noise lower too (around $6Hz$ by construction). That difference could be the reason for the disparity in influence more than the nature of the neurons, and while it could be interesting to investigate, it is not in the scope of this study and does not change the main results as the categorization was only use as a tool to visualize the data. This opens the way to a flexible modeling framework of internal subpopulations, whose precision can be adapted to the most significant level of description, depending on the context and the questions asked. This is a first bottom-up step towards a coarser description of the system, and

hence, may guide reliable modeling attempts at larger scales.

We have also established that not only this structure matters, but also its interaction with instantaneous finite-size fluctuations of the noise and the time evolution of the *global* dynamics. These are all constitutive of the observed behaviors, and none can be neglected to understand them.

Also, our results showed that, for the AdEx network, there exists a time window, characterized by a high variance across noise realizations, during which it is possible to reverse the behavior by applying an appropriate stimulation. The use of a stimulus to interrupt a seizure has been applied in the past in the case of absence seizure (Rajna & Lona, 1989). These results have been used as bases of computational studies at the scale of the EEG (Taylor et al., 2014). Computational work on the response of a network model to stimuli to disrupt seizure-like activities has shown the importance of the precise timing of the stimulation (Anderson, Kudela, Cho, Bergey, & Franaszczuk, 2007). Then, the use of electrode stimulation has been developed in rodents (Pais-Vieira et al., 2016). These different approaches have been implemented, including deep brain stimulation, vagus nerve stimulation (Boon, Cock, Mertens, & Trinka, 2018) and magnetic stimulation (Ye & Kaszuba, 2019). However, experimental recordings of the response to stimuli do not allow us to understand the mechanisms of large populations of neurons. Indeed, even if progress in calcium imaging or in multi-electrode arrays has made it possible since this last decade to record a large number of neurons simultaneously, we do not yet have access to the exact structure of the network they constitute. The study presented here is thus a proof of concept, based on a specific network model.

Finally, we also found that it is possible to "control" the propagation of the seizure by appropriate stimulation in a given time window. We think that this constitutes not only an important prediction of the model, but also a potential important possibility of treatment of some types of intractable focal epilepsies. This prediction could be tested in future modeling work at the mesoscopic scale, with realistic connectivity between the focus and neighboring areas. Such a model could be used to test the hypothesis that appropriate stimulation in areas adjacent to the focus may prevent the propagation of the seizure.

Perhaps the most exciting perspective is that the same paradigm could be used experimentally to control seizures. This would require a system to detect the onset of the seizure in the focus, and another system to deliver appropriate stimuli in adjacent areas. Such a system could be applied to experimental models of focal seizures, to evaluate if such a paradigm could revert the propagation – and thus generalization – of the seizure. This could be another way of controlling seizures, not by suppressing the focus, but by making sure that the paroxysmal activity does not propagate.

# Chapter 3 : Dynamical properties of self-sustained and driven neural networks


## Summary

This computational study aimed at unraveling the differences between two apparently similar systems : Driven and Self-sustained networks. We showed that later had a first Lyapunov exponent (FLE) an order of magnitude higher than the first, which translated in a tendency to have a higher response to perturbations. We also validated those networks results on a specific mean-field model.

In this work, I designed the experiment, did the simulations, the analysis, and I was the main contributor behind writing the paper.



Publication : (Bouté & Destexhe, 2023), preprint available in biorxiv:
`https://www.biorxiv.org/content/10.1101/2023.07.15.549166v1`

# Dynamical properties of self-sustained and driven neural networks

Jules Bouté and Alain Destexhe

Paris-Saclay University, CNRS, Institute of Neuroscience (NeuroPSI), Saclay, France

**Abstract**

In the awake brain, cerebral cortex displays asynchronous-irregular (AI) states, where neurons fire irregularly and with low correlation. Neural networks can display AI states that are self-sustained through recurrent connections, or in some cases, need an external input to sustain activity. In this paper, we aim at comparing these two dynamics and their consequences on responsiveness. We first show that the first Lyapunov exponent (FLE) can differ between self-sustained and driven networks, the former displaying a higher FLE than the late. Next, we show that this impacts the dynamics of the system, leading to a tendency for self-sustained networks to be more responsive, both properties that can also be captured by the mean-field model we used. We conclude that there is a dynamical and excitability difference between the two types of networks besides their apparent similar collective firing. The model predicts that calculating FLE from population activities in experimental data could provide a way to identify if real neural networks are self-sustained or driven.

## 9 Introduction

Neural networks are a core subject of analysis when it comes to modeling the brain. They can behave in vastly different ways, and we will focus here on the so-called Asynchronous Irregular (AI) state as defined (with the other types of behavior) in (Brunel, 2000b) and first observed in neural networks in (Amit & Brunel, 1997; van Vreeswijk & Sompolinsky, 1996; Vreeswijk & Sompolinsky, 1998). This means neurons fire without any significant coherence between each other and that single neurons would not have a regular pattern of activity.

That type of activity is similar to that observed in awake animals (Matsumura et al., 1988; Steriade et al., 2001; Destexhe et al., 2003; Lee et al., 2006), including primates and humans (Dehghani, Peyrache, Telenczuk, Le Van Quyen, et al., 2016) and therefore represent a basic type of activity of prime importance. This is why it is useful for networks to be able to reproduce it, and how it is an interesting approach

when testing common models.

Different models of single neuron and networks allow to reproduce these dynamics, such as the AdEx model (Brette & Gerstner, 2005a; Naud et al., 2008a), where it was shown that AdEx networks can display AI states (Destexhe, 2009b), which will be the focus of this study. The well-known Hodgkin-Huxley model (Hodgkin & Huxley, 1952a), which can also display AI states (Carlu et al., 2019), which we use here for comparison.

While the global activity can be reproduced well with such models, they can also have a large range of different behaviors as shown for example in (Naud et al., 2008a), and a change of parameters within the single neurons models or the network model can lead to vastly different dynamics. Although some of those dynamics can be obvious to differentiate (such as a change in the average firing rate that one would adapt to reproduce specific brain data, or reactions to specific inputs that could also be modulated), there are other sets of parameters that would appear very similar.

A good example is the existence of two different types of networks : Driven networks, that need an external drive such as in (Brunel, 2000b; Zerlaut et al., 2017) and Self-sustained networks, that just need to be started to have an activity such as in (Vogels & Abbott, 2005; Destexhe, 2009b).

Those two types of networks appear very similar if we overlook the need for an external drive, and the two of them could even be use with an external drive. Therefore, if they appear so similar, would those two types of networks behave the same way ? Can those two categories be meaningful and predictive of different dynamics altogether ? This are the questions we will try to answer in this study.

To do so, we use tools from dynamical systems (Eckmann & Ruelle, 1985; Ruelle, 1989) applied to computational neuroscience (Faure & Korn, 2001; E. M. Izhikevich, 2007; Cessac, 2009), and in particular the Lyapunov Exponents (Cessac, 2009; Ruelle, 2009) (as defined in (Wolf et al., 1985; Eckmann et al., 1987)), which we apply here to the two types of networks. We further probe the networks by studying their responsiveness to external synaptic inputs, as in (Yger, El-Boustani, Destexhe, & Frégnac, 2011; Zerlaut et al., 2017), allowing us to see how such network states could be used to process or transmit information in the brain (Zerlaut & Destexhe, 2017) or generate pathological activity such as epilepsy (Depannemaecker et al., 2022).

Finally, we will compare our findings with a specific mean-field models of the two network types, following (El-Boustani & Destexhe, 2009; Zerlaut et al., 2017; Volo et al., 2019), allowing us to see if the concepts and properties of Driven and Self-sustained networks are equivalent to that of the mean-field models we used.

# 10 Methods

## 10.1 Neural network model

We now present the model of single neurons that we use, and how said single neurons are linked together to form a network.

It is useful to point that we took parameters such that our networks would be Asynchronous Irregular, as defined in the Introduction.

### 10.1.1 AdEx model

To simulate a neural network, we fist need to simulate single neurons. We decided to focus on the Adaptative Exponential (AdEx) Integrate and Fire model (Brette & Gerstner, 2005a; Naud et al., 2008a). In the following, unless otherwise stated, we focus by default on the AdEx network.

Each neuron in the AdEx network is described by Eq.(39) and Eq.(42) as follows:

$$C\frac{dV}{dt} = g_L(E_L - V) + g_L\Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) - w + I_{syn} \tag{39}$$
$$\tau_w \frac{dw}{dt} = a(V - E_L) - w$$

With

$$I_{syn} = g_E(E_E - V) + g_I(E_I - V) \tag{40}$$

And

$$\frac{dg_{E/I}}{dt} = -\frac{g_{E/I}}{\tau_{syn}} \tag{41}$$

When the membrane potential crosses a threshold, a spike is emitted, and the system is reset for a given refractory time:

$$\text{if } V \geq V_D \text{ then } \begin{cases} V \to V_R \\ w \to w + b \end{cases} \tag{42}$$

Then, to create a biological networks, we simulate two kind of neurons, that we call populations : excitatory and inhibitory neurons. As the name suggests, when the first kind of neuron will spike, it will increase the chance of spiking of other neurons, while for the second kind it will decrease that chance. We simulate a total of 10 000 neurons, 2000 being inhibitory - that we will call FS for Fast Spiking neurons - and 8000 being excitatory - that we will call RS for Regular Spiking neurons. Those neurons will be based on the same model, but with different parameters. If we only give one parameter value, it means

it is the same for the two populations.

In order to influence other neurons, they need to be connected to each other. We choose a random Erdős–Rényi one way connection of 5%, so every neurons is on average connected to 400 excitatory neurons and 100 inhibitory neurons, and receives input for the same amount of neurons. Multi connection from one neuron to another is allowed, albeit unlikely, and self-connection is not allowed.

On top of those neurons, we will often use an external excitatory input from 8000 excitatory Poisson "neurons". The corresponding excitatory input will be used as a drive (see 11.1) or as a perturbation (see Section 11.3). The strength of that drive/perturbation will be given in term of the average firing rate of the Poisson neurons (for example, 2Hz).

In our networks, $C_m = 200pF$, $\Delta_T = 0.5mV$ for the FS population and $2mV$ for the Rs population, $a = 0nS$, $b = 0pA$ for the FS population and $\tau_w = 1ms$ for the FS population. $E_E = 0mV$, $E_I = -80mV$ and $\tau_{syn} = 5ms$.

As we use various networks with different single neuron parameters, the rest of thge parameters, namely $g_L$, $E_L$, $V_T$, $\tau_w$ and $b$ for the RS population, $V_D$ and $V_R$ will be given in sect.11.1.

### 10.1.2 Hodgkin-Huxley model

Although the primary model we used is the AdEx model, we also studied the Hodgkin-Huxley (Hodgkin & Huxley, 1952a) model, that we will call "HH" model.

$$C_m \frac{dV}{dt} = g_l(E_l - V) + g_K n^4 (V - E_K) + g_{Na} m^3 h (V - E_{Na}) + I_{syn} \tag{43}$$

with gating variables (in ms):

$$\begin{aligned}
\frac{dn}{dt} &= \frac{0.032(15. - V + V_T)}{(exp(\frac{15. - V + V_T}{5.}) - 1.)}(1. - n) - 0.5exp(\frac{10. - V + V_T}{40.})n \\
\frac{dh}{dt} &= 0.128exp(\frac{17. - V + V_T}{18.})(1. - h) - \frac{4.}{1 + exp(\frac{40. - V + V_T}{5.})}h \\
\frac{dm}{dt} &= \frac{0.32(13. - V + V_T)}{(exp(\frac{13. - V + V_T}{4.}) - 1.)}(1 - m) - \frac{0.28(V - V_T - 40.)}{(exp(\frac{V - V_T - 40.}{5.}) - 1.)}m
\end{aligned} \tag{44}$$

Apart from the single neuron model, HH networks function the same way as AdEx networks described previously, including eq.(40) and eq.(41) that are identical.

Although no artificial reset is required, we also have a specific $V_D = -10mV$ to detect the spike.

As previously, some parameters are the same for all networks.$C = 200pF$, $G_L = 10nS$, $E_L = -65mV$, $V_T = -47mV$ for the FS population and $-46mV$ for the RS population. $G_K = 6000nS$, $E_{Na} = 55mV$

and $E_K = -90mV$ . Finally, as for AdEx, $E_E = 0mV$, $E_I = -80mV$ and $\tau_{syn} = 5ms$. $G_{Na}$ is a specific parameter that will be given in sect.11.1.

## 10.2 Mean-Field

We used a mean-field models of AdEx networks, using the model defined in (Volo et al., 2019). This mean-field model was originally based on a Master Equation formalism developed for balanced networks of integrate-and-fire neurons (El-Boustani & Destexhe, 2009). This model was first adapted to AdEx networks of RS and FS neurons (Zerlaut et al., 2017), and later modified to include adaptation (Volo et al., 2019). This latter version corresponds to the following equations (using Einstein's index summation convention where sum signs are omitted and repeated indices are summed over):

$$T\frac{\partial \nu_\mu}{\partial t} = (F_\mu - \nu_\mu) + \frac{1}{2}c_{\lambda\eta}\frac{\partial^2 F_\mu}{\partial \nu_\lambda \partial \nu_\eta} \tag{45}$$

$$T\frac{\partial c_{\lambda\eta}}{\partial t} = \delta_{\lambda\eta}\frac{F_\lambda(1/T - F_\eta)}{N_\lambda} + (F_\lambda - \nu_\lambda)(F_\eta - \nu_\eta) + \frac{\partial F_\lambda}{\partial \nu_\mu}c_{\eta\mu} + \frac{\partial F_\eta}{\partial \nu_\mu}c_{\lambda\mu} - 2c_{\lambda\eta} \tag{46}$$

$$\frac{\partial W}{\partial t} = -W/u_w + b\nu_e + a(\mu_V(\nu_e, \nu_i, W) - E_L) \ , \tag{47}$$

where $\mu = \{e, i\}$ is the population index (excitatory or inhibitory), $\nu_\mu$ the population firing rate and $c_{\lambda\eta}$ the covariance between populations $\lambda$ and $\eta$. $W$ is a population adaptation variable (Volo et al., 2019). The function $F_{\mu=\{e,i\}} = F_{\mu=\{e,i\}}(\nu_e, \nu_i, W)$ is the transfer function which describes the firing rate of population $\mu$ as a function of excitatory and inhibitory inputs (with rates $\nu_e$ and $\nu_i$) and adaptation level $W$. These functions were estimated previously for RS and FS cells and in the presence of adaptation (Volo et al., 2019).

At the first order, i.e. neglecting the dynamics of the covariance terms $c_{\lambda\eta}$, this model reduces to:

$$T\frac{d\nu_\mu}{dt} = (F_\mu - \nu_\mu) \ , \tag{48}$$

together with Eq. 47. This system is equivalent to the well-known Wilson-Cowan model (Wilson & Cowan, 1972), with the specificity that the functions $F$ need to be obtained according to the specific single neuron model under consideration. These functions were obtained previously for AdEx models of RS and FS cells (Zerlaut et al., 2017; Volo et al., 2019) and the same are used here.

## 10.3 Lyapunov exponent algorithm

Considering the way we simulated the networks and mean-field, with differential equations, and the apparent differences in term of activity due to subtle differences, we chose to analyse the simulations with a dynamical systems lens, meaning we will study the trajectory of the dynamics on the phase space : the space of the dynamic of the different variables. To do so, we mainly focus on one of the primary tools of dynamical system analysis : Lyapunov exponents, that we will hereafter write LE.

LE are often (Eckmann & Ruelle, 1985; Pikovsky & Politi, 2016) introduce as a way to observe the stability of specific points of the system : fixed points, points in which the differential equations goes to 0, inducing an absence of evolution of the trajectory. Positive exponent means the point is unstable : a small perturbation will just grow and goes away from the fixed point. Negative exponents mean the point is stable : small perturbation will just decrease and come back to the fixed point. In multidimensional systems, there is one LE per dimension, meaning that, when look at the phase space, there could be a mixture of stable and unstable directions from a fixed point.

While useful, fixed point analysis is not the only thing that can be done with LE. A generalized version of this analysis allows to compute the stability of a whole system (Pikovsky & Politi, 2016) (or at least of a specific attracting object withing it), which is what interested us. Here, we would considered system that are bounded, meaning that if there is some unstable directions due to positive exponents, they would not go to infinite : the perturbation would just grow, until the two path of the original and perturb trajectory are completely uncorrelated. This is called deterministic chaos (Strogatz, 2019). Chaotic systems have, on appearance, similar properties to random, noisy ones, but they are entirely deterministic which allows for a whole different kind of analysis and predictive power. Knowing if a system is chaotic or not, is an important outcome of the estimation of LE, as at least one positive exponent is a defining feature of deterministic chaos (Wolf et al., 1985; Eckmann et al., 1987). It was also shown that many systems in the brain have properties consistent with chaotic systems (Korn & Faure, 2003).

Finding LE requires different algorithms, depending on the situation, from simple to complex one, especially when the goal is to find the complete spectrum and the associated direction (Pikovsky & Politi, 2016). When the differential equations are available, the basic idea behind the algorithm is to do some perturbation, let the system evolve, and check if the distance between the original and the perturb point has grown or shrunk. The logarithm of this distance will be the exponent. Of course, this method should be done a lot of time to converge to a good value, as there will be some differences in different regions of the system. As can be guessed, this only allows to compute the largest LE, that we will call FLE for “First Lyapunov Exponent". We will rank each LE from higher to lower. If one wants to obtain the rest of the spectrum, it is usually requires to go to an orthogonal subspace from the direction of the first exponent, in order to cancel its effect, and to start over again for each exponent. For more detail, see

(Pikovsky & Politi, 2016), as finding more than the FLE is outside the scope of this work.

We said previously that understanding if a system was chaotic or not was of prime importance, which is why this kind of analysis was not restricted to analysing systems for which the equations were known : algorithm were created to compute the LE (and especially the FLE, as it was the required one to determine if a system is consistent with chaos) from time series (Wolf et al., 1985; Eckmann et al., 1987), in order to apply it to real data.

Those algorithms work basically the same way as what was explained before, except that nearest neighbors are used instead of small perturbation. From then, the trajectory of the original point and the neighbor are observed for a certain amount of time steps in order to compute the change of distance between them. Finding the right amount is important : too many, and if the system is chaotic, all correlation between the two trajectory will be lost, meaning the distance would be meaningless. Too short, on the other hand, would not leave enough time for the trajectory to evolve, and would result in an increasingly longer computational time, on top of adding more potential errors. This is why we always tested two time steps, written as "Dt", to ensure the effect we found would be stable within a certain range. As before, we would take the logarithm of that distance that would have either shrunk or grow, to compute the FLE. It is again required to repeat the process and do an average on many times to converge to a meaningful value. The complete algorithm that was used is given in the supplementary section.

It is important to note that those algorithm works very well to find positive LE (not necessarily only the first one), but suffer from important bias when they try to compute negative ones (Wolf et al., 1985; Eckmann et al., 1987). In our case, only the FLE was positive, and the second one would be either 0 or negative, which is why we stopped our analysis in computing only the FLE of the systems (or the whole spectrum when fixed points where considered, in the case of the mean-field we used, as it would only be an analytical computation).

# 11 Results

We wanted to analyse the system using all variables, but it proved to be impossible due to their sheer number, as all tested algorithms relied on a distance metric (generally to find nearest neighbors) that always broke due to the curse of dimensionality.

Therefore, we needed fewer variables, which is why we chose variables that were the most relevant for the system, on top of being the ones used to simulate a mean-field of said system, allowing for a great connection between two different representations of the same phenomenon. Two obvious choices were the average firing rates of the excitatory and inhibitory population, which gives an idea of the average behavior of the system, and the last one is the average adaptation variable, only present in the excitatory population, which is a specificity of the AdEx model, and therefore an important feature to represent.

For HH, we will choose the other 6 variables : n,h and m for of the inhibitory and excitatory populations. It is worth noting that the first two variables are of a different kind from the last ones : the adaptation (or mn,h and m) is an intrinsic part of the description of the system, having its own dynamical equations, while the firing rates are product of the system, when the membrane potential reaches a threshold.

We start by describing the self-sustained and driven types of networks, then we compute their Lyapunov exponents, and next their responsiveness to external inputs.

## 11.1 Driven and self-sustained networks

Neural networks can vary in term of the model of single neuron or the type of connectivity as shown previously, but we want to add another difference : what we call the type of networks, which can be driven or self-sustained.

### 11.1.1 AdEx

Driven networks are the networks that require an external (here Poissonian) input to produce an activity (see Fig.5(A-B)). As most parameters will work to create driven networks, they are also the easiest to obtain (if we allow an external drive).
Self-sustained networks on the other hand only needs an initial “kick" to start, and will, as the name suggest, have a self-sustained activity after (see (see Fig.5(C-D)).

Both types of networks are interesting for different reasons : driven networks might be more realistic, as a small network is never cut out of the rest of the brain, and has no specific reason to sustained itself. On the other hand, self-sustained networks allows for an easier analysis of their dynamics, as there is no external input and most of all no sources or randomness (through the poissonnian input) that will affect them. Self-sustained are more difficult to obtained than driven : on top of requiring some fine tuning (although an important range of parameters is compatible with it), it also requires specific noise realizations (so a specific random connectivity and a specific kick), as many noise realizations will just die quickly (in less than a second, with a kick of 100ms).
It is important to note that, while some change of parameters and study of the dynamics of the system gives good insight on how to obtain a self-sustained network, it is beyond the scope of this study to prove they were actually self-sustained forever and it was only checked that they would have a seemingly stable activity for longer than the typical length of simulation we needed (more than 20s), as even if it is only a transient phase, its lifetime would be big considering the size of the network (Tél & Lai, 2008). While the self-sustained networks do not require it, we sometimes add a drive to them for the sake of comparison with the driven system.

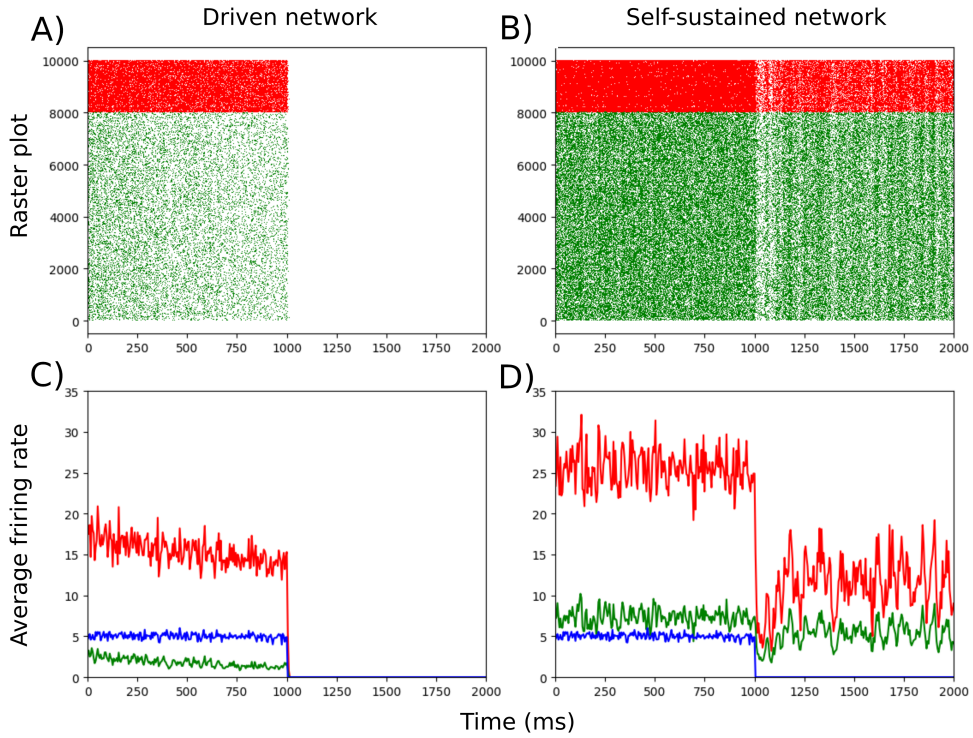


Figure 26: Illustration, for an AdEx network, of driven networks (A) and B)) and self sustained networks (C) and D)) with an external Poisson drive that stops at 1000ms. A) and C) represent raster plots, with green being RS neurons and red being FS neurons. B) and D) represent the average firing rate, with the same colour code and blue being the Poisson input. As can be seen, when said Poisson input stops, the driven networks immediately stop any activity, while the self-sustained reduce but maintain its activity.

**HH**

We created 2 Driven and 2 Self-sustained networks, each with specific parameters for the single neuron model as we presented in sect.10.1.1. It is interesting to note that to obtain that change of macroscopic regime, we did not change the connectivity or any "macroscopic" variables, but only the local, single neurons ones. We give the values in the next table. If the value is different for the FS and RS population, we give the two of them in the same cell, as in $FS/RS$, if it is the same, we only give one value.

| Driven | 1 | 2 |
|---|---|---|
| $g_L$ | $15nS$ | $10nS$ |
| $E_L$ | $-65mV$ | $-65/-70mV$ |
| $V_T$ | $-50mV$ | $-50mV$ |
| $\tau_w$ (RS) | $500ms$ | $150ms$ |
| $b$ (RS) | $60pA$ | $100pA$ |
| $V_D$ | $-47.5/-40mV$ | $-47.5/-40mV$ |
| $V_R$ | $-65/-55mV$ | $-65mV$ |

Table 2: Parameters of the AdEx single neurons to produce Driven networks. Parameters not presented were given in sect.10.1.1. Parameters that are different between the two Driven networks are given as "$X/Y$".

| Self-sustained | 1 | 2 |
|---|---|---|
| $g_L$ | $15nS$ | $18nS$ |
| $E_L$ | $-65mV$ | $-60mV$ |
| $V_T$ | $-49/-52mV$ | $-50/-52mV$ |
| $\tau_w$ (RS) | $500ms$ | $100ms$ |
| $b$ (RS) | $60pA$ | $60pA$ |
| $V_D$ | $-45mV$ | $-47.5/-40mV$ |
| $V_R$ | $-65/-55mV$ | $-65mV$ |

Table 3: Parameters of the AdEx single neurons to produce Self-sustained networks. Parameters not presented were given in sect.10.1.1. Parameters that are different between the two Self-sustained networks are given as "$X/Y$".

### 11.1.2 HH

We obtained the same two types of networks for HH networks, as can be seen if Fig.27. Here we can see that the firing rates look very similar to the AdEx networks, albeit being a bit higher due to a choice of parameters, especially for the self-sustained HH network (while it could have been possible, in theory, to have lower firing rates, finding the correct parameters to obtain a self-sustained activity is a difficult task, so those parameters were kept).

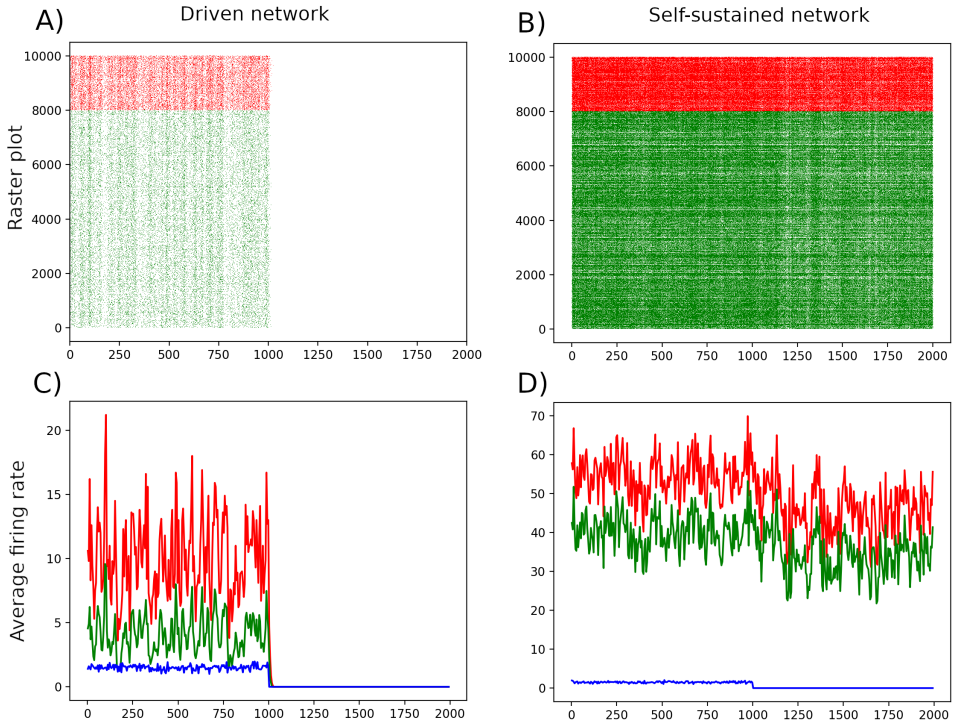


Figure 27: Illustration, for an HH network, of driven networks (A) and B)) and self sustained networks (C) and D)) with an external Poisson drive that stops at 1000ms. A) and C) represent raster plots, with green being RS neurons and red being FS neurons. B) and D) represent the average firing rate, with the same colour code and blue being the Poisson input. As can be seen, when said Poisson input stops, the driven networks immediately stop any activity, while the self-sustained reduce but maintain its activity.

As previously, we have different parameters for the different networks. Here, only one parameter changed,

and we only did one Driven and one Self-sustained network.

For the Driven network, we used $G_{Na} = 80000nS$ for the FS population and $40000nS$ for the RS population.

For the Self-sustained network, $G_{Na} = 20000nS$ for both populations.

**Mean-field**

With the mean-field we used, it was also possible to obtain Driven like and Self-sustained like activity, as can be seen in Fig.28 in the sense that with some parameters, the activity would go to 0 without external drive for some networks, while it would go to some non-zero values for some other. Due to the very different dynamical nature of this mean-field, as it has attractive fixed points instead of chaos, as we will see, it is hard to know if the Driven and Self-sustained networks are of the same type as the corresponding networks.

We can see some oscillations after a change of state (at the beginning for each network simulation, and when we stop the external input for the Self-Sustained network). This is most likely an artefact, due to this mean-field being created to simulate steady states, and not particularly the transient state when there is a change. This is why we will only consider the steady states values.

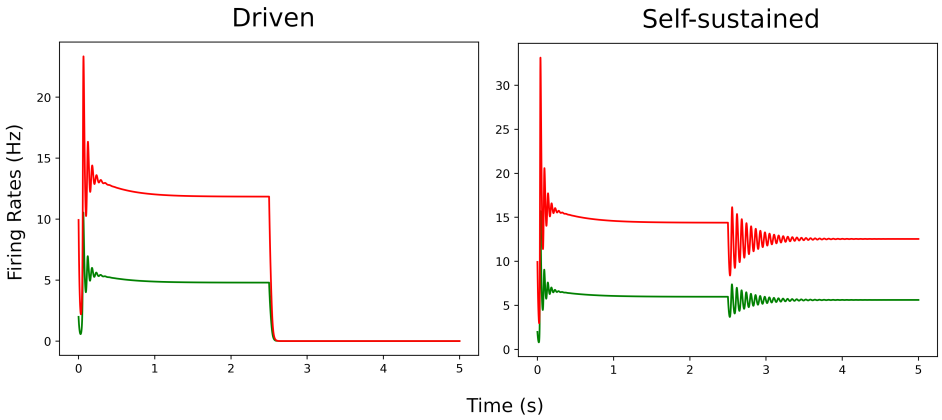


Figure 28: Driven (left) and self-sustained (right) mean-field, with fixed external input at 1.5Hz for 2.5s and then no input for the last 2.5s.

## 11.2 Lyapunov exponents comparison between Driven and Self-sustained systems

Using the algorithm described earlier, we computed the FLE on a dynamical system made from the 3 dimensions described previously (8 for HH). Usually, the algorithm requires to obtain the phase space using Takens reconstruction, but here we force it to operate in the three dimensions we constructed, as they 1) make sense to represent all the system (the normal algorithm often assume we only have access to partial information) and 2) make it easier to identify with the mean-fields (for AdEx as our mean-field

analysis was only performed for it, although the same would have been true for an HH mean-field similarly defined).

**AdEx**

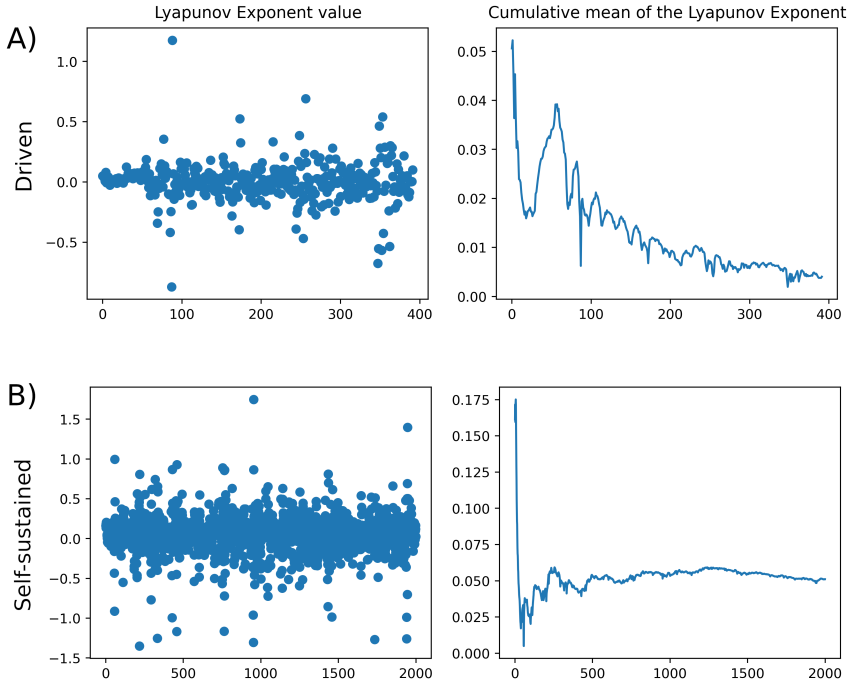


Figure 29: Example of FLE algorithm for typical self-sustained (up) and driven (down) AdEx networks. On the left, we have the values of the exponents at various steps. On the right, we have the cumulative mean.
The self-sustained converges to 0.05 and the driven converges to 0.002

Fig.29 is a good way to get a representation of how the algorithm works, and what it means to have a given LE : we can see it is really a converging value of the whole system, that varies from time to time (or more accurately, depending on when we take it in the system) but still converges, which is why there is some meaning in talking about the FLE of the different systems.

To explore the differences between Driven and Self-sustained networks, we computed the FLE for different conditions in Table.4

Here we can see that while the FLE is always fairly weak, it is also positive (meaning we have chaotic systems) and an order of magnitude higher when we compare Driven to Self-sustained networks.
Or at least, that is what happens when we compute it with the firing rates. As can be seen in the $2^{nd}$ part of Table.4, when we use the intrinsic variables of the membrane potential Vm, the result is very different, showing no significant differences between Driven and Self-sustained networks. Interestingly, it

| FR | Driven 1 | Driven 2 | Self-Sustained 1 | Self-Sustained 2 |
|---|---|---|---|---|
| Dt=10 | 0.01/0.003 | 0.007/0.0004 | 0.05/0.04 | 0.05/0.04 |
| Dt=20 | 0.004/0.003 | 0.006/0.002 | 0.04/0.04 | 0.05/0.07 |
| Vm | | | | |
| Dt=10 | 0.35/0.36 | 0.56/0.54 | 0.38/0.08 | 0.46/0.42 |
| Dt=20 | 0.35/0.31 | 0.41/0.41 | 0.20/0.21 | 0.33/0.32 |

Table 4: Values of the FLE for the different conditions. There are two parts : the first one is computed with the two mean firing rates (FR) and the mean adaptation, while the second part is computed with the two mean membrane potentials (Vm) and the mean adaptation. The column represent different networks, 2 driven networks with different parameters, and two similarly different self-sustained networks. The lines represent 2 different time steps used in the algorithm, to ensure some robustness. Finally, each condition was repeated with a different noise realization (including both the drive and the connections in the network).

also shows much higher LE. This shows that depending on the way we compute them, the exponents can be very different, and encourage us to use them more as a method of comparing than with their absolute values.

Table.5 shows similar results for a self-sustained network with an external drive, in order to see if the drive was the cause of the difference. As can be seen, results are similar to those for the "normal" self-sustained networks.

| | 1.5Hz 1 | 1.5Hz2 | 5Hz 1 | 5Hz 2 |
|---|---|---|---|---|
| Dt=10 | 0.05/0.05 | 0.06/0.06 | 0.06/0.05 | 0.06/0.06 |
| Dt=20 | 0.05/0.03 | 0.06/0.06 | 0.06/0.04 | 0.06/0.04 |

Table 5: Values of the FLE for the different conditions, computed from the FR. The column represent different self-sustained networks with different drives, 2 1.5Hz drives and 2 5Hz drives. The lines represent 2 different time steps used in the algorithm, to ensure some robustness. Finally, each condition was repeated with a different noise realization (including both the drive and the connections in the network).

**HH**

We also wanted to obtain the FLE for the HH networks, and as can be seen in Table.6 there is no difference between driven and self-sustained, which is why we did not investigate it further, focusing on the specificity of AdEx.

| FR | Driven | Self-Sustained |
|---|---|---|
| Dt=10 | 0.37/0.41 | 0.39/0.36 |
| Dt=20 | 0.26/0.27 | 0.31/0.26 |
| Vm | | |
| Dt=10 | 0.49/0.51 | 0.48/0.43 |
| Dt=20 | 0.34/0.36 | 0.37/0.33 |

Table 6: Values of the FLE for the different conditions for the HH model. There are two parts : the first one is computed with the two mean firing rates (FR) and the mean adaptation, while the second part is computed with the two mean membrane potentials (Vm) and the mean adaptation. The column represent different networks, 1 driven and 1 self-sustained. The lines represent 2 different time steps used in the algorithm, to ensure some robustness. Finally, each condition was repeated with a different noise realization (including both the drive and the connections in the network).

**Mean-Field**

Finally, we also computed the LEs for our AdEx mean-field. This analysis is a bit different, as we actually only have three equations and everything is continuous. It was therefore possible to compute the entire spectrum directly using the torch library in python (torch.autograd.functional.jacobian). The results for a Driven and a Self-sustained networks, with and without external drives, are given in Table.7. We can see that we indeed have an attractive fixed points in all cases, as all exponents are negative. On top of that, the $1^{st}$ and $3^{rd}$ are, interestingly, always the same. The only difference arise for the $2^{nd}$ exponents, which is always higher for self-sustained, although only the case without external input seems to be significantly different.

| Driven | $1^{st}$ exponent | $2^{nd}$ exponent | $3^{rd}$ exponent |
|---|---|---|---|
| Input = 0Hz | -2.00 | -66.67 | -66.67 |
| Input = 1.5Hz | -2.00 | -62.78 | -66.67 |
| Input = 3Hz | -2.00 | -65.96 | -66.67 |
| Self-sustained | | | |
| Input = 0Hz | -2.00 | -44.18 | -66.67 |
| Input = 1.5Hz | -2.00 | -62.15 | -66.67 |
| Input = 3Hz | -2.00 | -65.77 | -66.67 |

Table 7: Values of the FLE for the different conditions for our AdEx mean-field. There are two parts : the first one is for a driven network, and the second one for a self-sustained network. The column represent each LE. The lines represent different amount of external inputs (in Hz)

## 11.3 Responsiveness

Finally, we wanted to see if the difference in the FLE ($2^{nd}$ for this mean-field) would lead to differences in the response to an short external input between driven and self-sustained networks (mean-fields). As only AdEx showed differences, we only studied it for both the network and the mean-field.

To do so, we added an external poisson input, in blue in Fig.30. That added external input was done on top of the normal external drive for driven networks, and not in place of it, which is why we can see a non zero value everywhere that rise higher in the left of Fig.30 compared to the right where there is no external input apart from the perturbation.

We computed the response as follow :

$$Resp_{abs} = < FR >_{window} - < FR >_{base} \tag{49}$$

$$Resp_{perc} = \frac{Rep_{abs}}{< FR >_{base}} \tag{50}$$

There are two measures of responsiveness that we used here : the first one is the absolute response,

$Resp_{abs}$, and is the difference between the average FR during a specific window chosen to be inside of the perturbation times, on one hand, and the average FR outside of the perturbation : the base FR. While interesting by itself and enough to compare, said, similar networks (mean-fields) with different external perturbation strength, we needed to compare networks (mean-fields) that had different base FR, and it could be that it would influence the absolute value of the response. To avoid that issue, we also computed the percentage response $Resp_{perc}$, which is the $Resp_{abs}$ divided by the mean base FR. Those two measures will be shown in the next Tables.

**Network**

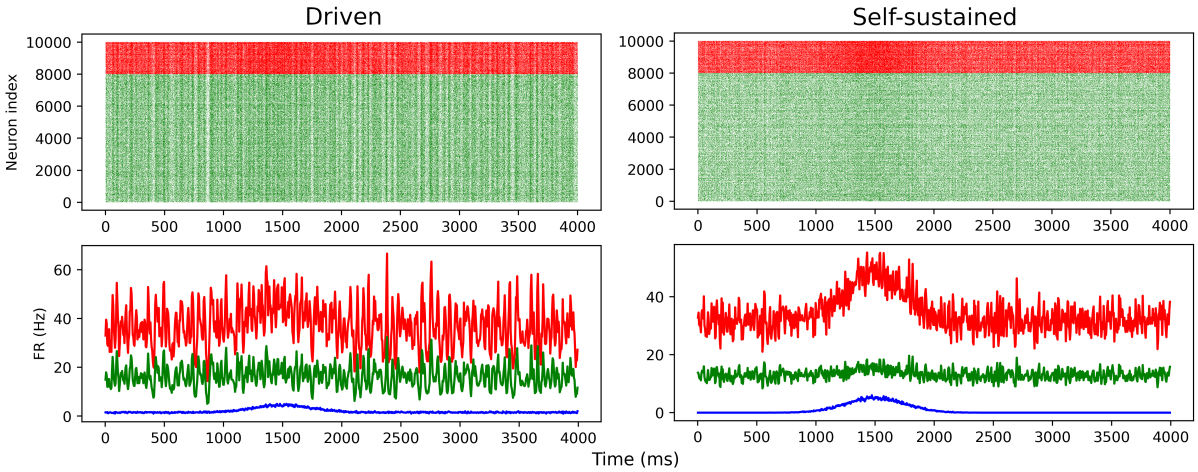


Figure 30: Response to a perturbation of 3Hz centered at 1500ms. Green represent the excitatory population, red is in inhibitory one, and blue is the external poisson drive. Up : raster plot. Down, average firing rates.Left part is for a driven network, right part for a self-sustained one.

Fig.30 shows Driven and Self-sustained networks responding to an external perturbation of 3Hz. The responsiveness $Resp_{abs}$ and the $Resp_{perc}$, as defined previously, were reported (for different networks and different perturbation strength) in Table.8. Here we can see that Self-sustained networks seem to respond more than Driven ones, although the difference is not big an variability within Driven or Self-sustained networks is already high. The results are not as clear as they were for the FLE, but there seem to be a clear tendency.

**Mean-field**

Fig.31 shows Driven and Self-sustained mean-field responding to an external perturbation of 3Hz. Contrary to Fig.30, there is no noise here and the dynamic is much simpler (as there is only 3 dimensions). Table.9 is identical to Table.8, apart from the absence of different noise realizations, as no noise were used in the mean-field. It presents a similar result to Table.8 : self-sustained mean-field do indeed appear

| Driven | Exc. FR 1 | Exc. FR 2 | Inh. FR 1 | Inh. FR 2 |
|---|---|---|---|---|
| Input = 3Hz | 0.67/0.84, 6.1%/7.5% | 0.11/0.11, 1.1%1.2% | 3.28/3.58, 12.4%/13.4% | 2.22/2.26, 9.8%/9.8% |
| Input = 5Hz | 1.13/1.23, 10.2%/10.9% | 0.17/0.17, 1.9%/1.9% | 5.39/5.60, 20.3%/20.8% | 3.65/3.71, 16%/16% |
| Self-sustained | | | | |
| Input = 3Hz | 8.29/8.25, 25.7%/24.8% | 11.87/11.12, 32.1%/34.4% | 1.27/1.28, 9.9%/9.5% | 2.79/2.35, 19.5%/19.4% |
| Input = 5Hz | 13.12/12.84, 40.3%/38.2% | 17.91/17.65, 47.5%/53.2 | 1.83/1.80, 14.2%/13.3% | 4.05/3.66, 27.9%/29.7% |

Table 8: Response of the network with different external currents (in lines) for 2 Driven and 2 Self-sustained networks and for the two neuronal populations (in columns) : excitatory and inhibitory. In each cells, we have two $Resp_{abs}$ for two noise realizations, followed by two $Resp_{perc}$, also for two noise realizations.

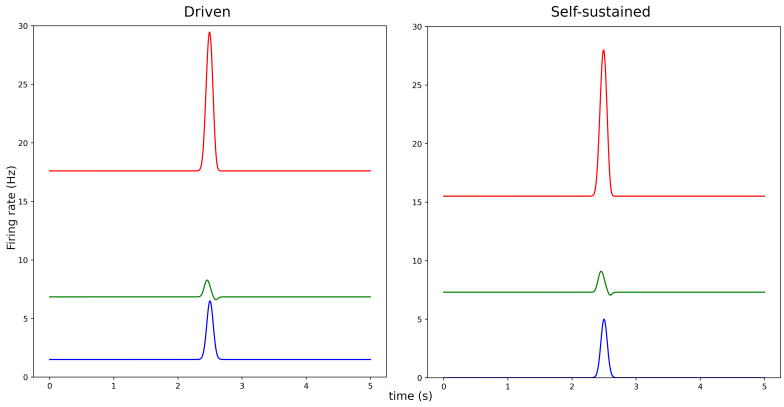


Figure 31: Response to a perturbation centered at 2.5s. Green represent the excitatory population, red is in inhibitory one, and blue is the external drive. Left part is for a driven mean-field, right part for a self-sustained one.

to have a higher response than driven ones.

| Driven | Exc. FR 1 | Exc. FR 2 | Inh. FR 1 | Inh. FR 2 |
|---|---|---|---|---|
| Input = 3Hz | 0.751,11.1% | 1.348,13.0% | 6.627,37.6% | 66.319,28.7% |
| Input = 5Hz | 0.869,12.7% | 1.745,16.8% | 10.228,58.1% | 9.694,44.0% |
| Self-sustained | | | | |
| Input = 3Hz | 0.994,13.6% | 1.771,16.8% | 7.160,46.2% | 7.026,35.4% |
| Input = 5Hz | 1.132,15.5% | 2.230,21.1% | 10.814,69.7% | 10.509,52.9% |

Table 9: Response of the Mean-Field with different external currents (in lines) for 2 Driven and 2 Self-sustained networks and for the two neuronal populations (in columns) : excitatory and inhibitory. In each cells, we have $Resp_{abs}$ followed by $Resp_{perc}$

It appears that, both for the AdEx networks and mean-fields, self-sustained ones (which had higher LE) have a higher response than driven ones.

# 12 Discussion

In this work, we analyzed two types of neural networks, self-sustained networks and driven metworks. We used Lyapunov exponents, an important tool in dynamical systems, to yield insight on the nature of their dynamics, and found differences between the two types of networks. We also investigated the responsiveness of such networks, and found that the self-sustained networks show a tendency to be more responsive than driven networks. These differences can be captured to some extent by the mean-field models we used. We discuss here these results and their implications.

One main finding is that the FLE is different for the two types of AdEx networks. It must be noted here that we did not technically compute the FLE of the network, as the AdEx networks have 40 000 dimensions (and the HH networks have even more). As it would have been difficult to use algorithms in such a high-dimensional space, partly due to the curse of dimensionality (Köppen, 2000), we restricted our use to a system made from the average values of the FRs and the Adaptation (or the other variables for HH). It is interesting to note that the FRs are directly correlated to a change of activity of the network, while the link between it and the Vm is more tenuous (due to the reset after the spike) and high or low values of Vm are hard to link with the global activity of the network. This could explain why we obtained interesting results with the FR but not with the Vm.

The positive FLE that was computed indicates we have chaotic system. It is interesting to note that this system evolves here (by construction) in a low dimensional space, and is likely to follow high dimensional chaos in the original 40 000 dimensions of the original system. But our main result, as show in Table 4 is that Self-sustained networks had a FLE roughly an order of magnitude higher than Driven networks. Of course, a major difference between Driven and Self-sustained systems, by definition, is that one of them has a constant external drive applied to it while the other does not. It seems obvious that this potential added drive would influence the dynamics one way or another, and that this influence would be found in the system made from the average values as well. It could have very well been that we observed only the difference due to the drive, that would have caused a decrease in the LE somehow. This is why we did simulations with the same Self-sustained networks as before, but with an external drive similar to the one we used for Driven networks. We then computed their FLE, as seen in Table. 5. If the difference between Driven and Self-sustained was due to the drive, then similarly as before, the FLE should be lower, but this is not the case : they are roughly of the same values, and higher drives do not seem to increase or decrease much of the FLE. This means that the difference we observed is not due to an added external drive, but due to the intrinsic dynamics and specific sets of parameters that create the different types of network.

Furthermore, different set of parameters were used to obtain the four networks, two Driven and two Self-sustained, and despite having only those parameters from the single neurons model changed, differences

between their FLE were observed between different types of networks, but not within the same type (or at least, not to the same extent).
In other words, two networks that look similar if one only look at their activity through the FR, are actually intrinsically different. The difference in the FLE means Self-sustained networks are more chaotic than Driven ones. This means they both behave differently, which can have an impact on how they interact with other systems are tool to measure them, or how they would change when parameters are changed.

As can be seen in Table.6, the result we just described is only valid for the AdEx model.
There are a few possible reasons explaining why we see a difference between the two types for AdEx but not for HH networks : it could be that the difference also exists but is not captured by our method of reconstructing a low-dimensional system, or that the difference is unique to AdEx networks, or that the difference uniquely does not work in HH networks but would work with, said, other integrate and fire models.
We want to emphasize that even if the result we show was only true for AdEx (which is not obvious yet), that would already be useful for studying AdEx itself, obviously, but also to study the dynamical differences between the networks created from AdEx and other models.

A second main result concerns the responsiveness of the two different types of networks. We found that self-sustained networks generally display a higher responsiveness compared to driven networks (Table.8). However, this does not mean all Self-sustained networks would have a stronger response than Driven ones : there is a huge difference between various networks of the same type, and numerous parameters could influence the response to a perturbation (Volo et al., 2019; di Volo & Destexhe, 2021). Here we merely show a tendency for a few specific examples. Future studies should scan the parameter space of such models in more detail to determine if this feature is universal.

It is important to note that the link between LE and responsiveness was shown in another study (di Volo & Destexhe, 2021), but the responsiveness was more related to the second Lyapunov exponent of the system. Interestingly, we found a similar correlation between responsiveness and second exponent in the mean-field model (Table.7 and Table.9). Such a relation between the LE spectrum and responsiveness, or more generally between stability and responsiveness, is also an interesting direction to explore in future work.

Finally, those results should be more analysed and considered carefully, because it is known that integrate and fire models LE cannot be computed that easily because of the intrinsic discontinuities that exist in the single neuron (Coombes, 1999), and how rich and complex the analysis can be because of it (Cessac, 2008; Cessac & Viéville, 2008). Here, the situation is slightly different considering we only take average values that should cancel the discountinuity effect, but a more careful analysis is required to confirm the

previous results.

## Acknowledgments

Research supported by the CNRS, Agence Nationale de la Recherche (ShootingStar project, AD), The European Union (H2020-945539, AD), and a PhD fellowship from the Initiativas Foundation (JB).

## Supplementary

The code showing the algorithms and simulations can be found at :

`https://github.com/JuBoute/lyapunov_neural_network`

# Chapter 4 : Dissipation, a tool to study similarities and differences in neural network

## Summary

In this chapter, we introduced a follow-up of the Lyapunov Exponents : the Dissipation. This measures allowed us to produce observations from the neurons scale to the network scale, and delivered some new similarities and differences between AdEx and HH neurons, on one hand, and Driven and Self-sustained networks, on the other hand. A special focus was put on the responsiveness of the networks, and intriguing links were found between it and the Dissipation for AdEx networks.
In this work, I participated in designing the concept (the use of dissipation), designed the "experiments" (how could we use the dissipation), and I did the simulations and analysis.

# 13 Introduction

In this section, we will present meaningful results we obtained on dissipations and that help us understand the differences between AdEx and HH, on one hand, and Driven and Self-sustained, on the other hand, in a new way.

Dissipation is scarcely used in dynamical system theory, although it is far from being unheard of (Ruelle, 1996; Gallavotti, 2004; Gaspard, 2020). Although dissipative system theory and the fluctuation dissipation theorem are occasionally used in neuroscience (Lindner, 2022; Deco, Lynn, Perl, & Kringelbach, 2023), the measure of dissipation itself appears very rarely.

As we said when we introduced it in section 3.4, the dissipation is a tool of dynamical system theory linked with the LE, as it is the sum of all LE.
While it is far from being as precise as all the exponents, or while it does not give the same information as the FLE that we used in the previous chapter (as the dissipation will not tell us if the system is chaotic, for example), it still provides interesting information about the system we are studying.

We will start by showing how the dissipation works in our systems, to get a feeling of it.
To do so, we will first present the dissipation of different systems, showing differences and similarities.
We will then show how the dissipation is directly related to the global dynamics of the system : how it measures how fast a system dissipates, as the name suggest, to a stable structure.
To finish this introduction on dissipation of our neural networks, we will focus on the specific case of Self-sustained networks and what they mean.

Then, we will move on to our main results. Dissipation, due to its nature as a "coarser" version of the LE, is also a macroscopic observable of the system, and not of average values as we had to do in the previous chapter. This means the dissipation could be related to the core dynamic of the system in a different way than the LE.

Our first investigation was related to the link between the dissipation and the average activity of the FR. This study allowed us to find more interesting differences between Driven and Self-sustained systems.

Then, in order to recall our two previous chapters, we checked the responsiveness of different networks and how it was related with the dissipation.

In general, we compared Driven networks of various drives and Self-sustained networks (which sometimes had drives added on them, and sometimes didn't). If nothing specific is said, it means we are representing a Driven network and all networks had similar results (at least in term qualitative results, quantitative differences could happen but were not deemed relevant).

# 14 Method

## 14.1 Models of neural networks

We use here the same models as before : AdEx (Eq.(39), Eq.(42)) and HH (Eq.(43), Eq.(44). On top of that, for the responsiveness part, we also used another model : the leaky integrate and fire, or LIF, which follows eq.(51) and eq.(52).

$$C_m \frac{dV_m}{dt} = g_L(E_L - V_m) + g_E(E_E - V_m) + g_I(E_I - V_m) \tag{51}$$

When the membrane potential crosses a threshold, a spike is emitted, and the system is reset:

$$\text{if } V_m \geq V_D \quad \text{then} \quad V_m \to V_R \tag{52}$$

As can be seen it is the same as 39 and 42, but without the added exponential and adaptation part. Actually, as we said previously, AdEx was build on integrate and fire models with some added complexity. The parameters are similar to the one presented in 2.3.3. As before, there are two added equations related to the synaptic conductances :

$$\frac{dg_E}{dt} = -\frac{g_E}{T_E} \qquad \frac{dg_I}{dt} = -\frac{g_I}{T_I}$$

## 14.2 States of the network

In section 2.5, we introduce different behavior of networks, and we then specifically used AI networks.

On top of those networks, we also represented Up and Down (UD) networks : as the name suggest they will alternate between very high and very low activity, which are associated with sleep (Destexhe, 2009b; Torao-Angosto, Manasanch, Mattia, & Sanchez-Vives, 2021). This is a complementary macroscopic state that we used to analyze if the response rules we found for AdEx network would generalize.

## 14.3 Driven vs Self-sustained networks

Following chapter 3 we also used the dichotomy of Driven and Self-sustained networks for both AdEx and HH network. As a reminder, Self-sustained networks only require an initial excitatory input to start their activity, and they then keep going without any inputs, making them completely deterministic (Fig.5, right). On the other hand, Driven networks will die without an external drive (Fig.5, left).

## 14.4 Dissipation of neural networks

We already introduced the dissipation and how to compute it previously in 3.4, so we will now show the equations allowing us to compute the dissipation of our different models. To do so and to ease notation, we present the dissipation for a neuron, knowing the total dissipation will be the sum of all the single neurons dissipation.

First, the easiest one, the LIF model, in Eq.(53)

$$Diss = -\frac{1}{C}(g_L + g_E(t) + g_I(t)) - (\frac{1}{T_E} + \frac{1}{T_I}) \tag{53}$$

Please remember that $g_{E/I}$ are variables depending on the inputs of the neuron, while the rest are constants. This means that the only variation of the dissipation for LIF models will be related to the input from other neurons, and the leaking of that added conductance. It is also worth noting that, as all constant and variable are positive, the dissipation of the LIF network is always negative by construction.

In the AdEx model the dissipation function takes the following form in eq.(54)

$$Diss = -\frac{1}{C}(g_L(e^{\frac{V_m(t)-V_t}{\Delta_T}} - 1) + g_E(t) + g_I(t)) - (\frac{1}{T_E} + \frac{1}{T_I}) - \frac{1}{\tau_w} \tag{54}$$

On top of what we said for the LIF network, here the AdEx dissipation also depend on the membrane potential Vm, which is an important change. We can also note that, this time, a positive dissipation is possible thanks to the exponential term.

Finally, in the HH model the dissipation function takes the form of eq.(55) :

$$Diss = \begin{cases} -\frac{1}{C}(g_L + g_K n^4(t) + g_{Na} m^3(t) h(t)) \\ -\alpha_n(V(t)) - \beta_n(V(t)) \\ -\alpha_m(V(t)) - \beta_m(V(t)) \\ -\alpha_h(V(t)) - \beta_h(V(t)) \\ -\frac{1}{C}(g_E(t) + g_I(t)) - (\frac{1}{T_E} + \frac{1}{T_I}) \end{cases} \tag{55}$$

Where we simplify the equation with $\alpha$ and $\beta$, as was done between eq.(44) and eq.(5).
Here the dissipation is a little bit more complicated, but we still have similar parts, like the leaking conductance and the synaptic conductance contribution. As for AdEx, the dissipation will depend of

the synaptic conductance and on the membrane potential. Now that we have introduce how we will compute the dissipation for single neurons, it is useful to mention something important. Mathematically, the dissipation is defined as the Divergence of the flow, which is a sum of partial derivative from each differential equations in our system. We represented it for one neuron, but of course, the system will just be the sum over all the neurons of the dissipations defined for each single neuron. As we have explained, to have the total dissipation one should integrates over the whole space, which corresponds to integrate over time (although we do not demonstrate the ergodicity of our systems here).

This means the instantaneous Dissipation is properly defined (or, at least, the Dissipation on a very short time scale, which is what we produce in our simulations). But what about the Dissipation over one neuron ?

Technically, it is not properly defined. Going back to the Lorenz system in 3.2, it would be as if we separated each of the 3 differential equations (14), which would be abusive and only give us a tendency for our behavior on those specific directions in the phase space. The most interesting and solid information is indeed the total dissipation, but we claim here that the dissipation per neuron is nonetheless interesting. While it should be manipulated more carefully, it still gives information about the reaction of that specific neuron, although it would be foolish to consider it isolated from the rest of the network.

Therefore, while it is necessary to be careful, we believe dissipation is a way to analyze the behavior of single neurons in a network and to connect it to the rest of the network's behavior.

# 15 Results

## 15.1 What values does the dissipation take ?

We start by plotting the average dissipation through time, for both AdEx (Fig.32 up) and HH ((Fig.32 down) Driven networks. The result are similar for Self-sustained networks, so we do not show them.

We can see the dissipation per neuron is 20 times stronger for HH than it is for AdEx.

It also appears that the dissipation is mostly stable, although there are some variations. The distribution through time does not reveal any specific tendencies.

There are some differences if a different drive is used, but it merely changes the average value : the stronger the drive, the stronger the dissipation, therefore we do not show them.

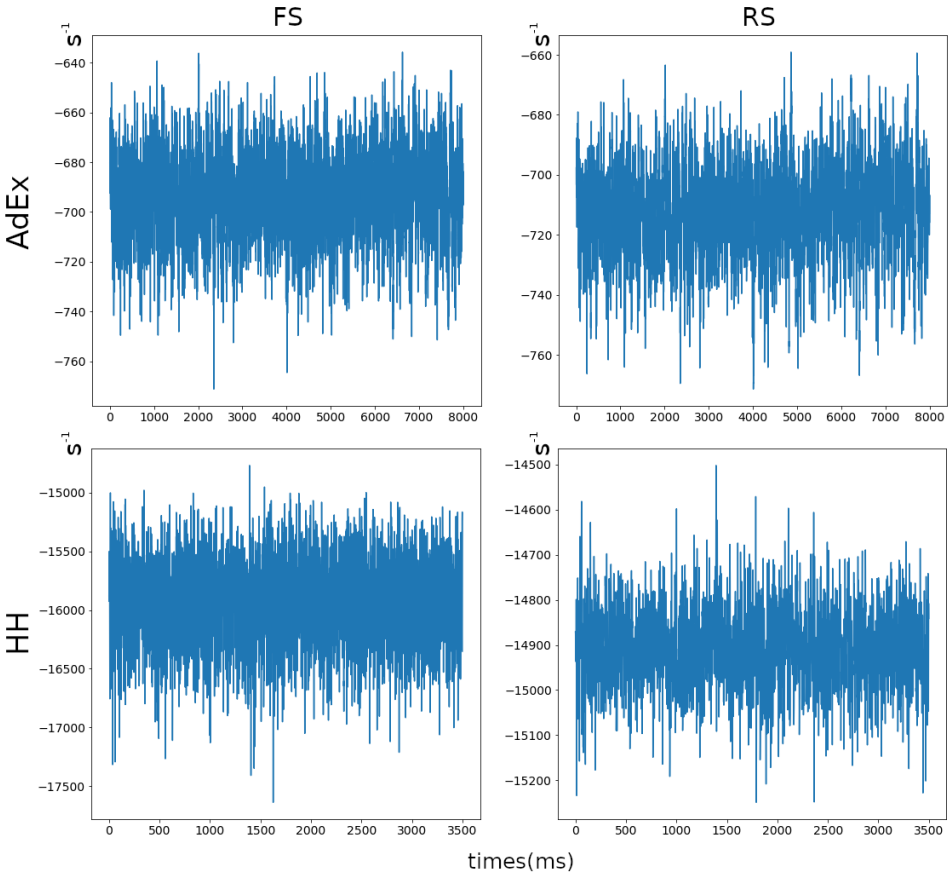


Figure 32: Mean Dissipation for AdEx and HH for a drive of 5Hz

We also plotted the distribution of the dissipation through the whole simulation per neurons in Fig.33. We can see that AdEx produce a mostly gaussian distribution, while for HH we have a skewed distribution, with a long tail toward stronger dissipation, which means few neurons can dissipate way more when they are excited enough.

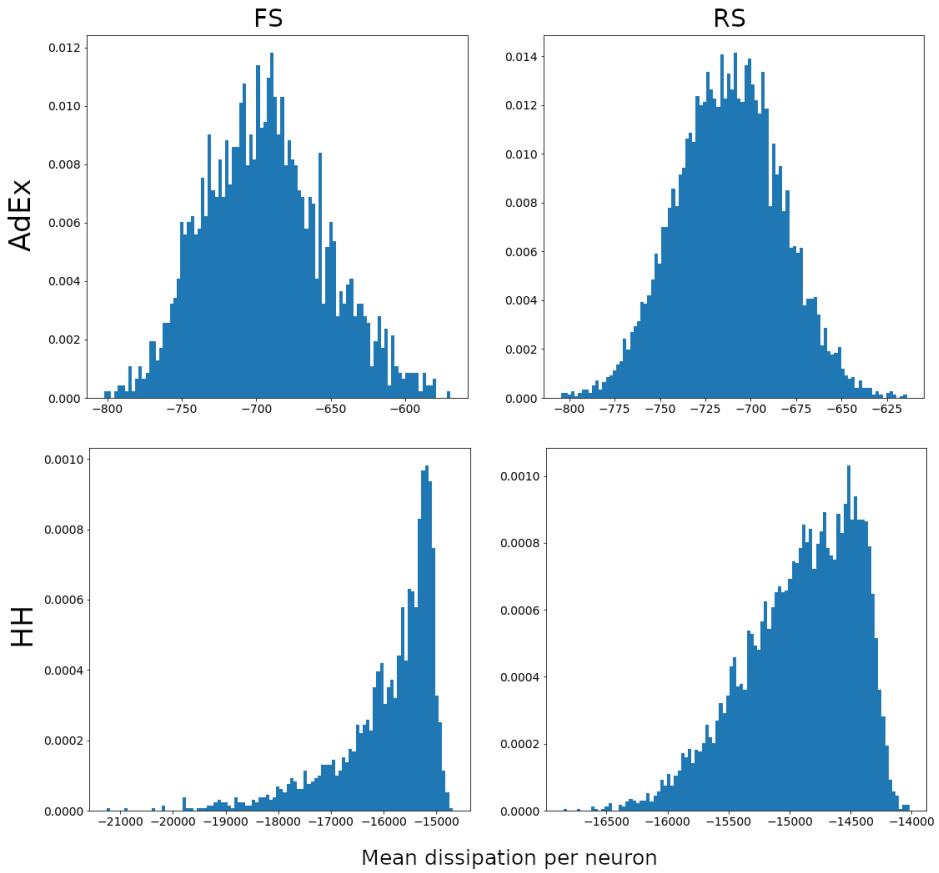


Figure 33: Distribution of the dissipation of the neurons for AdEx (up) and HH (down) Driven network

## 15.2 Transient time before the stable dynamics

Now that we have seen the typical values of the dissipation, we will see if it influences the time a network needs to stabilize.

For Driven networks, we here show the evolution of the dissipation at the initialization of the network (before it stabilize). We can see it takes about 1ms for HH (Fig.34 (down)) to stabilize, while it takes AdEx 20ms (Fig.34 (up)). For both of them, the dissipation starts weak (close to 0) and then grow stronger.

It is a bit counter intuitive : we could think the dissipation would be very strong when the system was out of equilibrium/the trajectory of the attractor, and then gets weaker. It could be that, as a magnet, the dissipation is stronger when the system is closer to the attractor.

Anyways, it appears the system stabilize more or less 20 times faster for HH networks than for AdEx ones, which is of the same range as what we saw in Fig.32, confirming what we said in the introduction with Fig.11: in its core, dissipation shows how quickly a system goes to its attractor.

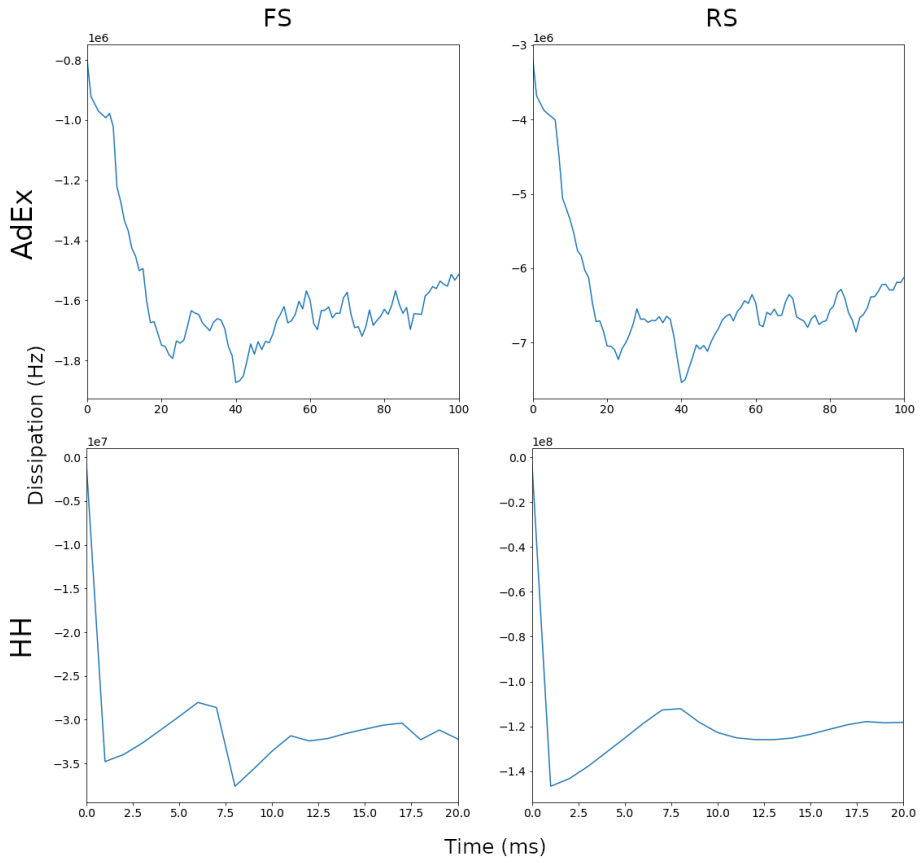


Figure 34: Dissipation at the very beginning of the simulation, during the so-called "transient time", before the networks reaches equilibrium, for an AdEx (up) and HH(down) network

## 15.3 Dissipation of Self-sustained activity

We observed some general truth on how the dissipation is a marker of the dynamics of networks created with two different models.

We will now observe the differences between Driven and Self-sustained networks, focusing on Self-sustained networks here. As previously discussed, the neuron and network parameters have been chosen such that the network only needs an initial kick to display some activity which seems stable over time.

In order to probe whether the system is dissipative, and how it is generally affected by the initial conditions, we start by initialising with various amplitudes and duration of the initial kick, and measure its average firing rate and dissipation over time (after having discarded a transient). See Fig. 35.

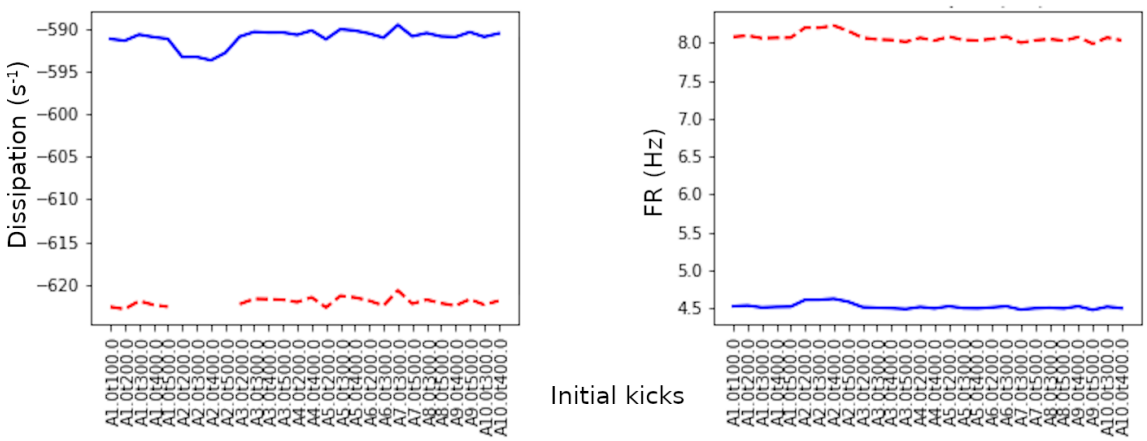


Figure 35: Global dissipation and firing rates of Self-sustained dynamics with various amplitudes and duration of initial kicks. A1 to A10 correspond to the amplitude of the kick, for values between $0.5Hz$ to $5Hz$ with an increment of $0.5Hz$. 0t correspond to the duration of the kicks, in $ms$, from $100ms$ to $500ms$. Blue lines are for the excitatory population, and red lines for the inhibitory one.

The very first result that comes across is the fact that the Self-sustained regime under investigation is globally dissipative. More interestingly, no matter which amplitude and time of initial kick is given to start it, the system will eventually converge to the same non-equilibrium state (demonstrated through same firing rate and dissipation).

This latter observation must be compared with what is obtained with a Driven network, with various levels of external drive, see Fig. 36. In this situation, both global firing rate and dissipation strongly depend on the level of input.

Also, it must be noted that, due to randomness of structure and noise, there are situations when the Self-sustained network does not manage to reach its Self-sustained state.

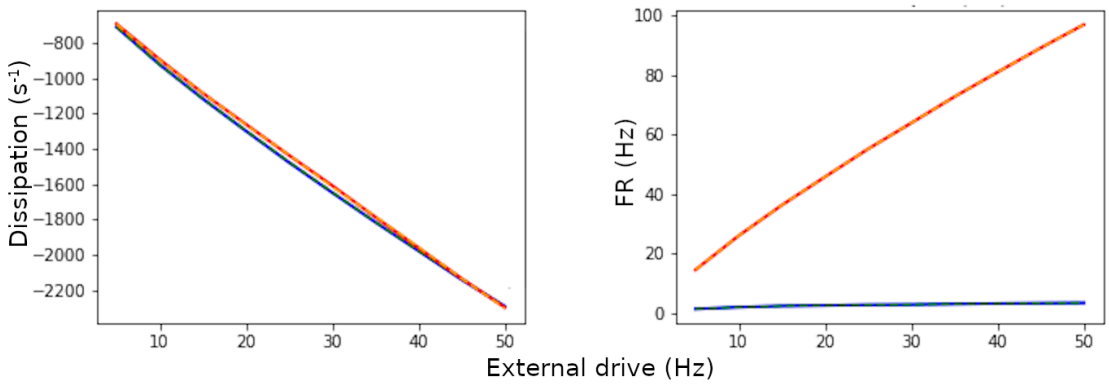


Figure 36: Global dissipation and firing rates of Driven network with various amplitudes of external drives. Blue lines are for the excitatory population, and red lines for the inhibitory one.

## 15.4 Link between dissipation and the FR

From this observation of Self-sustained systems, we wanted to compare them to Driven networks. We said previously that their activity, through their firing rates, appeared similar, but we wanted to dig that question. To do so, we computed the Pearson correlation between the dissipation and the FR.

| | AdEx | | HH | |
|---|---|---|---|---|
| | 1Hz | 5Hz | 1Hz | 5Hz |
| FS | 0.03, 0.07 | 0.14,0.13 | 0.19,0.21 | -0.07,-0.15 |
| RS | -0.23, -0.17 | -0.16,-0.17 | 0.18,0.21 | 0.56,0.53 |

Table 10: Values of the correlation between the dissipation and the FR for Driven networks of 2 drives (1Hz and 5Hz) and 2 models (AdEx and HH), for two noise realizations

| | AdEx | HH |
|---|---|---|
| | Self-sustained | Self-sustained |
| FS | -0.58, -0.60 | -0.42 |
| RS | -0.57,-0.60 | -0.67 |

Table 11: Values of the correlation between the dissipation and the FR for Self-sustained networks and 2 models (AdEx and HH) for two noise realizations

First, we can see that different realizations lead to very close results, so it is not a big factor here.
We found that there is a strong positive or weak positive or negative correlation for Driven networks, whether AdEx or HH, while there is a strong negative correlation for Self-sustained networks, whether AdEx or HH. It appears the type of the system (Driven, in Table.10 or Self-sustained Table.11) has more impact on its relationship between dissipation and FR than the model itself, or the type of population (RS or FS).

## 15.5 Responsiveness and dissipation

### 15.5.1 Introduction

Here, the goal is to evaluate how the network responds to a perturbation and what we can learn from the dissipation. We define as "response" the difference in FR between the basal FR and the FR during the perturbation, as we did in section 11.3. We also compare both Driven and Self-sustained network, and to compare them more easily, we put a drive in said Self-sustained network.

In the first part, we study the AdEx network.
We use 10 drives, from 2 to 20 (going by increment of 2), 3 amplitudes [2,5,10]Hz and 3 realizations.

On top of observing the plots, the result were fitted with a linear, quadratic and exponential functions, and we then checked the error of the fit (distance between the real value and the fit). Please note that said error is an absolute one, as it is a distance, which means it can only be compared with similar scales. Fortunately, the error is a difference of response, and said response is the same whether we represent it

as a function of the dissipation or as a function of the FR, or whether we compare the fit to a given perturbation (said, 5Hz for the RS population) to another.

It is however not the same scale between the two populations : FS goes from 2.5 to 15Hz while RS goes from almost 0 to barely 2Hz. An error of 0.1 around 1Hz is therefore 10 times bigger than one of 0.1 around 10Hz.

As it is, however, not the goal of this study to compare the error of the populations, the distance will be enough. This also means it would be inconsiderate to compare the errors from different amplitude of perturbation as they correspond to different typical response, the only things we can do is compare a given perturbation with another from a different fit.

On a side note, to analyse the figures, we have seen that the weaker the drive, the higher the response will be. Conversely, for dissipation, it means that set of points with high response correspond to low drive.

### 15.5.2 AdEx

We checked three different parameters : 1 of Up and Down (UD) (technically doesn't work for Self-sustained) and 2 of AI. All of them gave similar results. The simulations are 10s long with the 1s perturbation starting at 7s.

Let's begin by observing the response as a function of the FR. Note that it can be expected for the FR to give good predictions of the response, as they represent the "activity" of the network, and the perturbation is closely related to said activity.

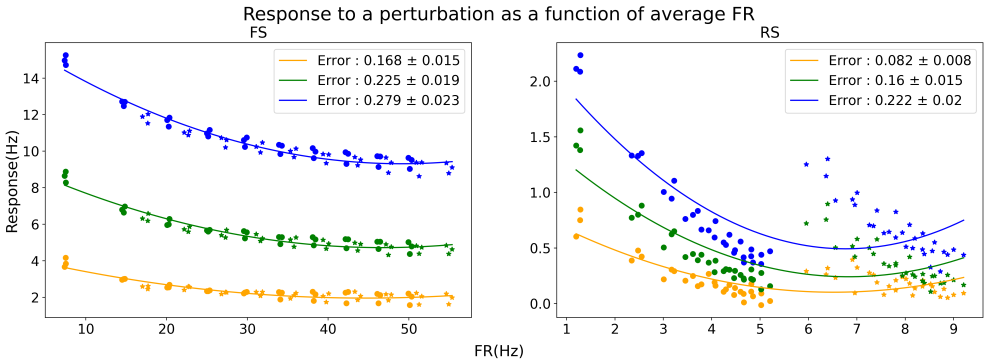


Figure 37: Response to a perturbation as a function of the FR for an AdEx network with AI behavior. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.
On top of those points, we show a quadratic fit. The error is computed by taking the average absolute difference between the prediction of the response from the fit given the value of the FR and the actual response for all points.

We can see in Fig.37 that, for FS neurons, the FR does indeed predict the response fairly well. The

error seem rather low for all fits (we only show quadratic fits here, as it was the best, especially for high perturbations) and we can observe a collapse between the Driven and Self-sustained networks : the FR prediction is only varying with the strength of the perturbation, not the type of the network. The RS population is different, as there is a clear separation between the two types of networks : the FR no longer predict things correctly with only the strength of the perturbation, it also required the type of the network.

Now, we can observe the response as a function of the dissipation.

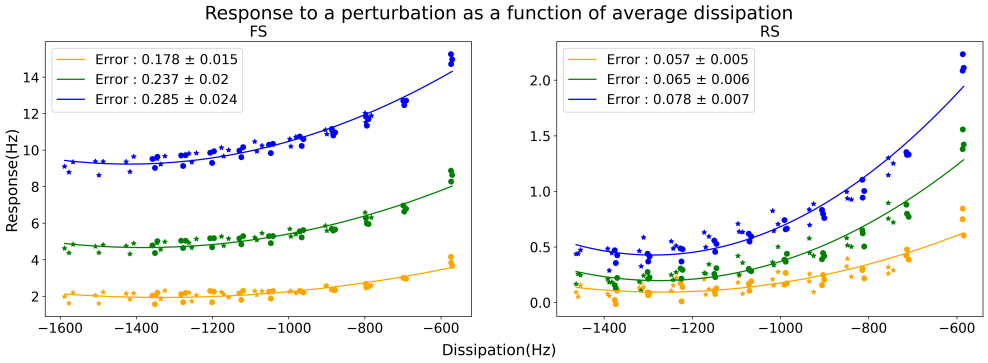


Figure 38: Response to a perturbation as a function of the dissipation for an AdEx network with AI behavior. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.
We show a quadratic fit. The error is computed by taking the average absolute difference between the prediction of the response from the fit given the value of the FR and the actual response for all points.

First we see the "collapse" between Self-sustained and Driven, that only depends on the amplitude of the perturbation. The difference with the FR is that this collapse works for both populations of neurons.

Again, we found that the best fit was the quadratic one. We can observe that the error does not seem to linearly follow the average value of the response for different perturbations.

We can see that the error is smaller than the prediction from the FR, with a Z-score that shows significant difference that becomes higher with the amplitude :

| | Perturbation strength | | |
|---|---|---|---|
| | 1Hz | 5Hz | 10Hz |
| FS | $Z = 0.47^{ns}$ | $Z = 0.45^{ns}$ | $Z = 0.17^{ns}$ |
| RS | $Z = -2.57^{*}$ | $Z = -5.66^{****}$ | $Z = -6.49^{****}$ |

Table 12: Z-score difference between the error of prediction from the quadratic fit of the FR and the quadratic fit of the dissipation for an AdEx network with an AI behavior. Negative values mean the dissipation has a smaller error than the FR.
$ns$=non-significant.
$* = p_{value} < 0.05$.
$**** = p_{value} < 0.0001$.

The same analysis was done for another AI state (meaning that different parameters where used), and while the dissipation part gave similar results, the FR part was different as can be seen in Fig.39

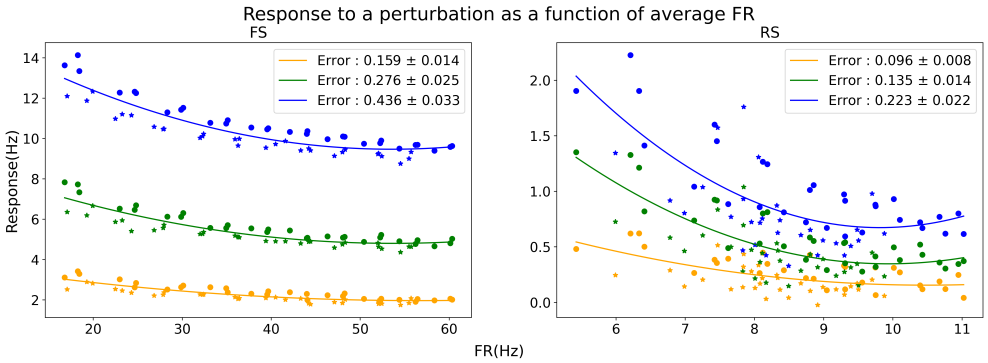


Figure 39: Response to a perturbation as a function of the FR for an AdEx network with AI behavior obtained with different parameters than Fig.37. Each point represent averages from a whole simulation. Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.
We show a quadratic fit. The error is computed by taking the average absolute difference between the prediction of the response from the fit given the value of the FR and the actual response for all points.

Here we can see that there is a sort of collapse between the Driven and Self-sustained networks that did not existed before. The Z-scores is therefore different, showing a weaker but nonetheless existing superiority from the dissipation.

| | Perturbation strength | | |
|---|---|---|---|
| | 1Hz | 5Hz | 10Hz |
| FS | $Z = -1.52^{ns}$ | $Z = -2.71^{**}$ | $Z = -3.19^{**}$ |
| RS | $Z = -0.77^{ns}$ | $Z = -1.58^{ns}$ | $Z = -2.13^{*}$ |

Table 13: Z-score difference between the error of prediction from the quadratic fit of the FR and the quadratic fit of the dissipation for an AdEx network with another AI behavior. Negative values mean the dissipation has a smaller error than the FR.
$ns$=non-significant.
$* = p_{value} < 0.05$.
$** = p_{value} < 0.01$.

Finally, a similar analysis was done with a different behavior of the network : an UD state.

Again, the dissipation is similar to what we observed before, with the collapse for both populations and a rather low error, and here the FR is even more divided than it was in the Fig.37, with otherwise similar results, suggesting the behavior of the network is of little importance for the response.

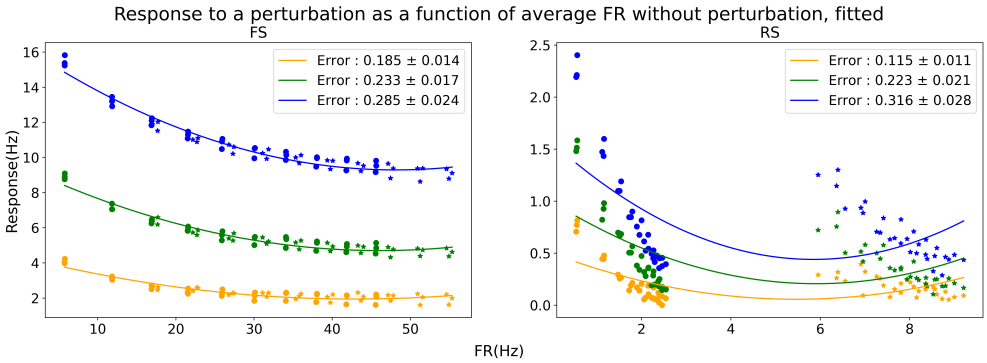


Figure 40: Response to a perturbation as a function of the FR for an AdEx network with UD behavior. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.
We show a quadratic fit. The error is computed by taking the average absolute difference between the prediction of the response from the fit given the value of the FR and the actual response for all points.

We have the following Z-scores :

| | Perturbation strength | | |
|---|---|---|---|
| | 1Hz | 5Hz | 10Hz |
| FS | $Z = 0.71^{ns}$ | $Z = 0.65^{ns}$ | $Z = 0.60^{ns}$ |
| RS | $Z = -4.52^{****}$ | $Z = -7.02^{****}$ | $Z = -7.62^{****}$ |

Table 14: Z-score difference between the error of prediction from the quadratic fit of the FR and the quadratic fit of the dissipation for an AdEx network with UD behavior. Negative values mean the dissipation has a smaller error than the FR.
$ns$=non-significant.
$**** = p_{value} < 0.0001$.

Which are very similar to the ones from Table.12, with the same behavior of having ns but positive z-scores for FS and very negative z-scores for RS, while Table.13 had always negative z-scores with many ns or weakly significant.

We can see that in all cases, the stronger the perturbation was, the better the dissipation predicted it compared to the FR.

All in all, it shows that dissipation is a good predictor of the response for the AdEx network, especially compared to the FR.

It's important to emphasized that FR already represent the kind of activity of the network that is measured with the response, and is therefore expected to be linked to it. It is therefore a powerful result showing that this new way of looking at networks can indeed gives deep understanding of its activity.

### 15.5.3 HH

We now turn to another model : HH.

HH network is harder to parameterize, specifically for the Self-sustained part, which means it was not possible to have FRs as similar as for AdEx. This could partially explain the results we will show, but does not explain everything.

Here, just one macroscopic behavior was tested, as it is not as easy (if not impossible) to have UD without the adaptation variable.

#### Total dissipation

The analysis and subsequent Fig.41 and Fig.42 are done as before, with the same different drives and amplitude of perturbation. As HH takes longer to simulate, only 4s were simulated, with a a perturbation at 2s.

First, let's check the FR :

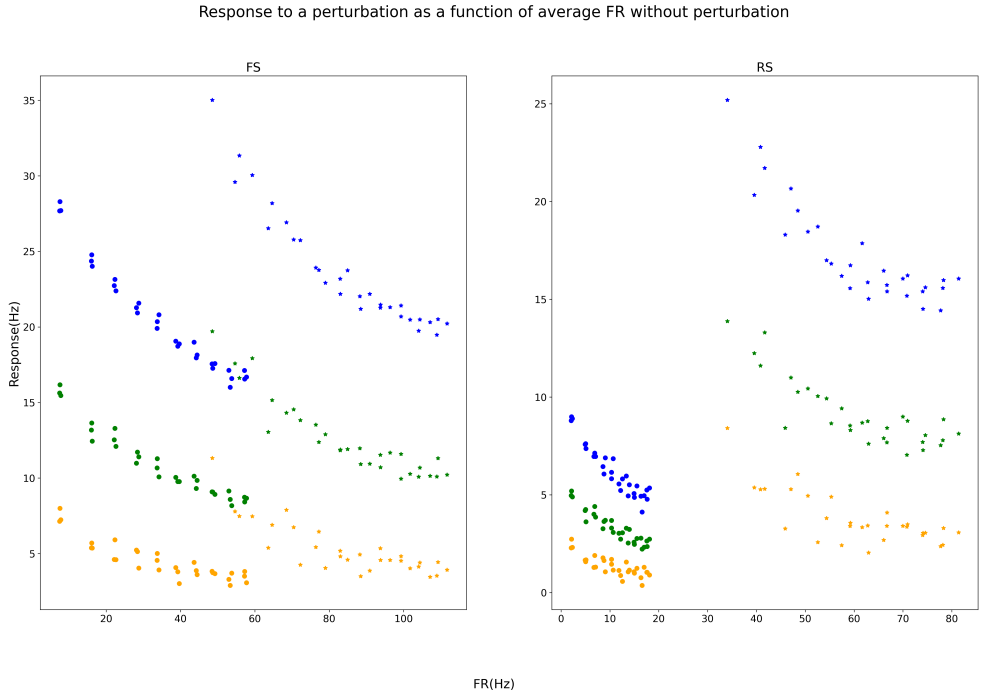


Figure 41: Response to a perturbation as a function of the FR for an HH network. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.

Here we can see a clear difference between the two types of networks, for the two populations. The

Self-sustained networks always respond more. They also have a higher FR for the two populations, which is due to the difficulty to parameterize them, as said previously.

Then, the dissipation :

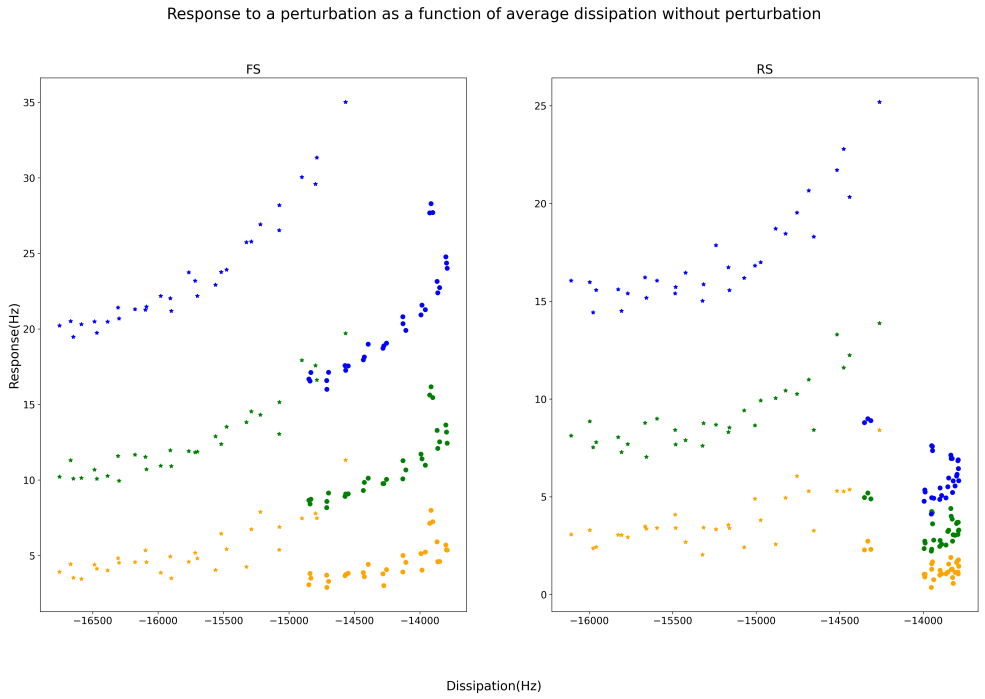


Figure 42: Response to a perturbation as a function of the dissipation for an HH network. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.

Unfortunately, the dissipation gives similar results as the FR.

While the Driven and Self-sustained networks followed lines regarding the perturbation and the drive, they were separated by their type, meaning no collapse were available here.

On top of that (for both the dissipation and the FR), we can see the Self-sustained is reacting less to the perturbation as its response is lower.

The effect we had for AdEx network does not seem to be present here.

We can notice that, for Driven networks, a same dissipation can give different responses (an effect that does not appear for the FR). Those responses belong to successive different low drives, that will indeed give response smaller and smaller, but sometimes an identical dissipation.

We already know from section 15.4 that Self-sustained system are negatively correlated to the FR, which means the stronger the FR, the stronger the dissipation. Here, it would appear that for HH, adding a drive change that correlation, while it does not for AdEx.

We performed the three fits again and found the quadratic fit is still the best, but it now gives fairly high errors, with small difference between prediction by FR and dissipation (none of them giving a significant difference).

It could be useful to check with different parameters, maybe trying to obtain higher FR for the Driven networks, as it is difficult to modify the Self-sustained ones.

**Detailled dissipation** We wanted to check if specific components of the dissipation (dissipation from $Vm$, $m$, $n$ or $h$, as the dissipation from the $G_{syn}$ is constant) were more predictive of the responsiveness, so we separated them and analysed them one by one. As we said for the single neurons, it is important to remember they are no longer the actual dissipation, but merely a tendency of the dynamical behavior of the network in one direction of the phase space.

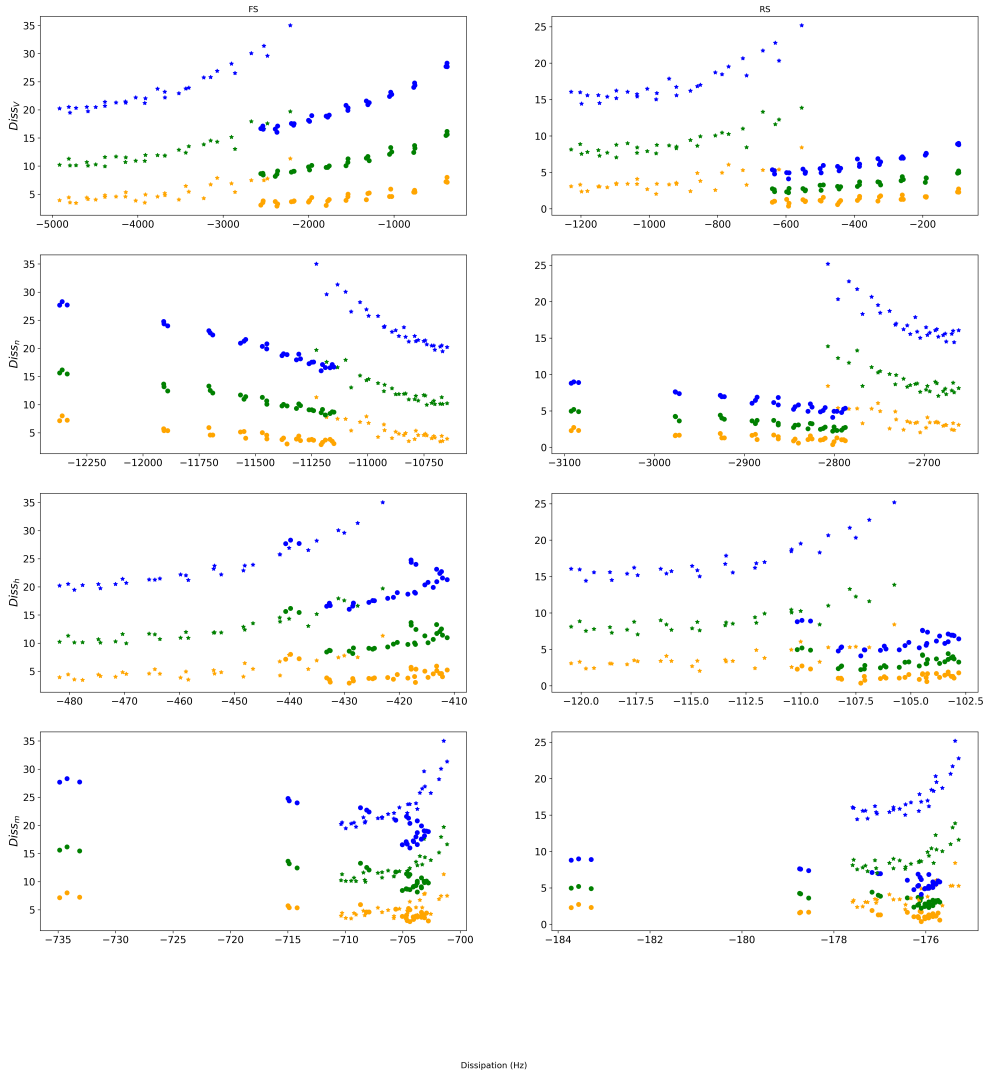


Figure 43: Response to a perturbation as a function of the different dimensions of the dissipation for an HH network. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.
The 4 lines represent the different dimensions of the dissipation : from $Vm$, $m$, $n$, and $h$ respectively.

Unfortunately, we can see in Fig.43 that no better collapse or prediction arose from this analysis, giving the same result as the total dissipation.

Still, we can see that the two responses for a dissipation only appears for $h$ and $m$, which produce the weakest dissipation, 8 times weaker than the one from $Vm$ and 20 times weaker than the one from $n$.

We can also see that $Vm$ and $h$ have a dissipation from the Self-sustained that is stronger than the ones

from the Driven. It becomes unclear if their is a difference for $m$, and it's actually the opposite for $n$. On the other hand, and while $n$ has the highest value, it has a smaller width (in absolute), than $Vm$ (2 to 3 times lower), which explains why the order of $Vm$ is the one we observe for the total dissipation.

It could also be that there are specific neurons that predict the responsiveness better, but the analysis should be done and the results are far from obvious.

All in all, dissipation is useful to predict the response for AdEx, but not so much for HH, which raised the question as to why, and also to know which one is the exception. It would be useful to test on a third model to answer that question.

### 15.5.4 LIF

We did the same analysis as previously with a Leaky Integrate and Fire (LIF) network, to see if it is the integrate and fire property that allows for the effect we observe on AdEx but not on HH.

First, we check the FR in Fig.44

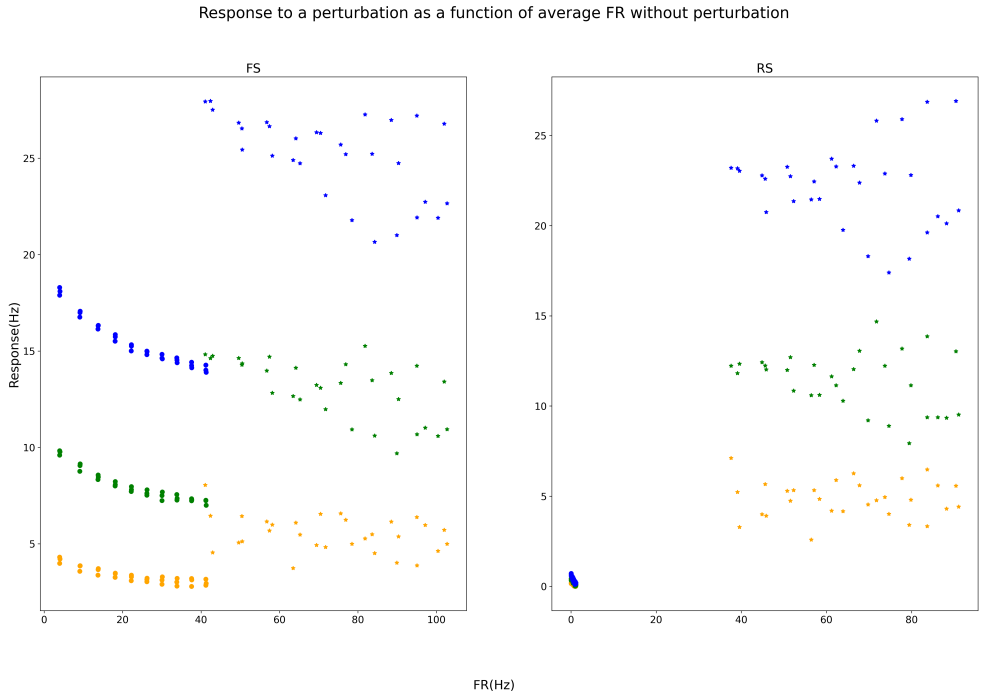


Figure 44: Response to a perturbation as a function of the FR for a LIF network. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.

Then, we do the same with the dissipation inf Fig.45

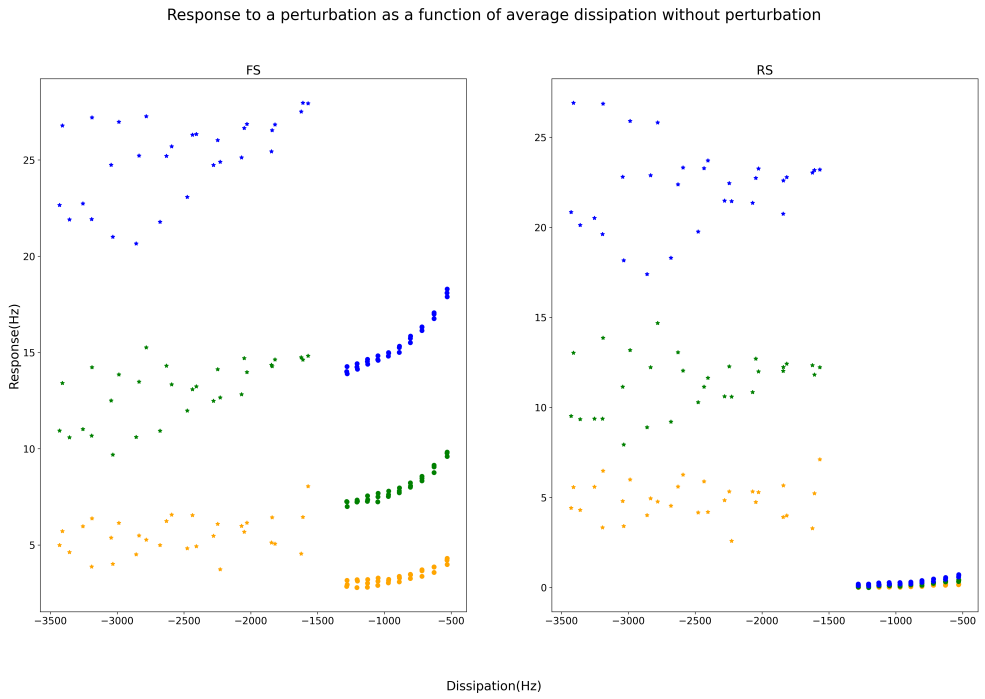


Figure 45: Response to a perturbation as a function of the dissipation for a LIF network. Each point represent averages from a whole simulation.
Colors indicate various amplitude of perturbation : 2Hz, 5Hz and 10Hz.
Circles represent Driven networks, while stars represent Self-sustained networks.
There are simulations with a drive from 2HZ to 20HZ, with a step of 2Hz, and each of those simulations were done with 3 noise and connectivity realizations.
The two columns correspond the FS and RS populations of neurons.

We have the same comments as for HH : there is no collapse between the two types of networks.

We can see nonetheless that the Driven FR are all concentrated in a small volume, while the equivalent dissipations still show more differences, which could mean dissipation allows for an easier differentiation between the various drive, but it is not really related to the response.

On top of that, the dispersion regarding the response for the Self-sustained (for both RS and FS) depending on the realization is way higher than for other models, suggesting that here the precise network connectivity or the noise play an important role, but only on the case of Self-sustained networks. Apart from that, we cannot really say there is the "doubling effect" we observed for HH Driven.

It could be that the adaptation is required, so it would be interesting to try another model with adaptation (like a modified HH with adaptation).

# 16 Discussion

First, we showed that the dissipation of networks created from different models (AdEx and HH) was different, HH being 20 times more dissipative than AdEx.

Then, we showed the direct implication of it : the HH is much faster at stabilizing to its stable manifold than AdEx. The result would be similar after a perturbation, and we in fact observed it (although not to the point of quantifying it) in the Annex.

Then, after those "global results" we started to observe the difference between Driven and Self-sustained networks, following what was done in chapter 3.

We showed that the Self-sustained system was dissipative, which is actually a surprise. These observations seem in contradiction with what is usually assumed in dynamical systems literature, in which, when a system is able to sustain without external drive, it cannot be dissipative in principle, and vice versa.
In Hamiltonian systems, the canonical conservative systems, the trajectories will remain during the whole simulation on an *energy shell*, defined by the initial conditions. However, in our situation, the initial conditions seem to play no particular role, except giving enough “energy" for the system to reach its sustained state. We thus find ourselves in the presence of a somewhat hybrid system : it is able to sustain some activity with no forcing, but has to converge to it, and it is constantly dissipating.

However, this apparent contradiction can be resolved by acknowledging the fact that the models we study are not totally situated in the “pure" dynamical systems framework : extra events are added, which are not described by the equations of motion, such as spiking events, instantaneous jumps and resets. These events not entering the continuous formulation of the global time derivative of the system, they are not taken into account in the dissipation function. However, they do act on the global dynamics, which is the real substrate of the dissipation function, which implies that it is indeed affected by them, but in a rather indirect way. In other words, while the continuous dynamics is mostly dissipative, the discrete events tend to maintain it, thus hinting at the possibility of establishing links between network activity and dissipation.

This link is actually our next results, and one of our main ones : the correlation between the FR - representing the global activity of the network - and the dissipation differs between Driven and Self-sustained networks. It is mostly close to 0 or positive for Driven networks, without much clear differences between AdEx and HH (although those differences would probably become clearer if an extensive analysis was done). On the other hand, it is always negative (and much more than the eventual negative cases from Driven networks) for Self-sustained networks, and again there does not seem to be much differences between AdEx and HH.

Dissipation can be seen as the system containing itself to avoid exploding. It appears Self-sustained systems link that with the FR, although the causal relation is unknown (all we know is that the stronger the FR, the stronger the dissipation).
Exploring what causes this link and how it differs from Driven systems could help us analyse the differences

between those two types.

But those results really shine when put in perspective with the results of chapter 3, where we showed that the Self-sustained systems had a FLE an order of magnitude higher than the Driven ones, but only for AdEx.
We wondered whether the difference between the two types was absent in HH, if it was that it was not captured by the FLE, or if the algorithm was not working well for various reasons. The result we find here is an indication toward the later : we show that there is indeed a difference between the types of network **independent** of the model, so it cannot be the first assumption, and we showed that this difference is linked to the dissipation, the sum of the LE, which raise questions on whether the FLE could really be unaffected.
This result exemplify why the dissipation should be use on top of the LE analysis, as it allows to find differences that were absent with the LE analysis alone.
More work would be needed to truly understand the link between the FR and the dissipation, but it seem to be a promising area of research.

Finally, the links between the FR - the activity - and the dissipation made us wonder how the dissipation could be link to the response of a perturbation - the change of activity. This question was important for many different reasons : 1) as we just said, it makes sense as it is link to the change of activity, 2) dissipation is also linked to the "recovery" of the system after a perturbation, 3) perturbation were the main subject of analysis in chapter 2 we presented (although we studied it in a very different way, as we only have healthy responses where the networks "control" the perturbation and let it die here) and 4) it was also a measure we made in chapter 3, showing how it was influenced by the FLE difference between Driven and Self-sustained networks.

Here we showed that, for AdEx, the average dissipation was a better prediction of the response of the network than the average activity itself. This is not at all trivial, considering the activity is, obviously, not a bad way to predict a change in said activity, and considering that the dissipation is not made to predict the response. We can even see that the prediction can follow a rather simple equation, although we did not tried too many fits.
The interesting part for the rest of our analysis is that we no longer see differences between Driven and Self-sustained (although it is useful to note that we used a Self-sustained with a drive to compare perturbation perfectly, but results from chapter 3 showed that the drive did not change the difference between Driven and Self-sustained).
Here, the only thing that matters was the value of the dissipation, and as we said in the first part of this chapter, there was not obvious differences between the dissipation of Driven and Self-sustained networks. This means that, while different in types, and while it could still very well be that Self-sustained networks

have a tendency to respond more than Driven ones, taking similar dissipation make this difference no longer meaningful.
This is interesting as it shows limits to the results we found in the previous chapter. To explain it, our main hypothesis is that Self-sustained networks have a tendency to be more dissipative than Driven ones, but an extensive search on those types of networks would be required to show it.

On the other hand, we showed that this link between responsiveness and dissipation no longer holds for HH. We find once again the difference between the two models we found in the previous chapter : we have some dynamical observable (FLE or dissipation) that produce some interesting effects for AdEx, and no specific effects for HH (although it is important to remember that an effect was also found for HH, as we said a few paragraphs ago). The reason behind those differences could be the same one as for the previous chapter.

The difference, though, is that we suggested previously to try other networks to see if they would behave similarly to AdEx or HH networks, and we did it here with a LIF network (although we did not test its FLE).
We were thinking that the difference was due to the integrate and fire part, but we can see that this was not the case (at least when it comes to the special link between response and dissipation, but we think it would be the same for the FLE).

Then we turn to our other hypothesis : AdEx is a special network that reacts specifically to dynamical observable, contrary to other networks, whether it's with its FLE or the response to a perturbation.
It could be that the difference is due to the adaptation part, which allows for a wide range of behavior impossible to have for other networks, but this needs to be investigate further, and the dissipation seems like a promising way to do so.

It should be stressed out that dissipation has rarely been used in computational neuroscience, and those results should therefor be taken with a grain of salt, considering it is unclear if said dissipation was correctly applied, or wether it was possible to consider "single neuron" dissipation, for example. A possible limitation arises from the discontinuity of the integrate and fire AdEx model, and could be responsible for the differences we observe here, similarly to the issues we could have had with the Lyapunov exponents in chapter 3.

# Chapter 5 : Global discussion and conclusion

We will now go back to the main concepts we developed in the three previous chapters, and tie them together in order to show how apparently similar networks can actually have a different dynamic, and how to study them.

**Summary of the results**

To begin with, we will make a quick summary of the main methods and results of the last 3 chapters.

First, in chapter 2, we studied the effect of a paroxysmal input on a network, differentiating between two cases : when the input "propagates", meaning the network switches to a pathological activity where all neurons spike as much as possible ; and when the input is "controlled", meaning the network shows a simple increase in activity without showing pathological behavior.
To study those differences, we used three models : AdEx, CAdEx, and HH. The first two models are very similar by construction and gave similar results, so we will only talk of AdEx and HH.
We showed AdEx networks had a bistability, switching between healthy and pathological behavior apparently randomly, while HH seemed to mostly have a healthy reaction, albeit sometimes with very strong responses, but more as a continuous change with the strength of the input.

We investigated the switch and the different states for AdEx, finding that the chances to have one or another could change due to the strength of the input and its shape, but also the connectivity of the network.
To understand the differences further, and as the analysis of the whole network did not seem to be fruitful enough, but the analysis on the single neuron scale was too big, we created a coarse description of the network by grouping neurons based on their inhibitory connections. This coarse description helped us understand the propagation better, showing how the activity of those groups would change right before what could be called a bifurcation (and, conversely, how it did not change when the network controlled the input).
This analysis right at the moment of the bifurcation allowed us to find a critical time for propagating cases for which another small input could be added to reverse the propagation.

Then, in chapter 3, we studied the difference between Driven and Self-sustained networks.
Self-sustained networks are autonomous networks : after an initial kick, a balance between the excitation and the inhibition is achieved, and the network keeps as stable AI activity.
Driven networks, on the other hand, require an external drive to have an activity, if they don't have it the inhibitory population becomes too strong and the kills all activity quickly.

As previously, we used AdEx and HH models of single neurons to create our networks, and both types could be found in them. The two types of networks appear very similarly if we only look at their FR, so

we had to find another way to analyze them : the LE.
We reconstructed a system from the average values of the two FR and the Adaptation, and computed the FLE from it.

We found that for AdEx networks, the FLE of Self-sustained networks was an order of magnitude higher than the one for Driven ones.
On top of that, it appears the response to an external perturbation had a tendency to be stronger for Self-sustained networks than for Driven ones, a difference we attribute to their difference in FLE, as it makes sense that a more chaotic network would react stronger to a perturbation.
We also validated those results on a mean-field made to reproduce the average values of the network we used to create the system we presented earlier. We only did that analysis for AdEx networks : HH ones did not show any differences between the FLE of Driven and Self-sustained networks, therefore we did not push the rest of the investigation on them.

Finally, in chapter 4, we investigated how dissipation could help us understand the dynamics of neural networks better, focusing on the difference between AdEx and HH, on one hand, and the one between Driven and Self-sustained networks, on the other hand.

We showed that the absolute value and the distribution per neuron were different between AdEx and HH models, but similar between different types of networks (as the FR was similar, although we can also have similar FR for AdEx and HH networks).
Then, we showed that the correlation between the dissipation and the FR would drastically change between Driven and Self-sustained networks, independent of the model of single neuron used, which confirmed that differences between those two types are also present in HH networks.
Finally, we investigated the relation between the dissipation and the response of the network to a perturbation, showing an interesting link between the two of them for AdEx but not for HH networks.

**Interpretations**

To begin with, now that we look at all chapters at once, we can see that in chapter 2 we chose to measure the membrane potential to capture the activity of the network, while on chapter 3 we chose the FR. In both cases, we justify those choices claiming the other one would not work, which could appear as a contradiction.
Both of those measures, as we said, are related to the activity of the network, which is the spike activity.

The main difference is the core nature of those observables : the Vm is an intrinsic part of the system, as it is always the first of the differential equations that define the single neuron. The FR, on the other hand, is always an external measure we do after the simulation, as it refers to the time for which we have

a spike. Another important distinction is that Vm is by nature a continuous variable at all scales, while the FR, if only taken at the neuron scale, is discrete by definition. It only acquires a continuous like state when we take averages of a lot of neuron, and it is questionable which amount would be enough. Typically, it is uncertain the small subgroups we developed in chapter 2 would be big enough to assimilate the FR to a continuous measure, which was required for the rest of the analysis. This, by itself, already justify the use of Vm in chapter 2.
But then, why not using it in chapter 3 too ?

The interpretation of the FR is actually very easy. If it is high during a certain time window, it means the network has a lot of neurons firing at that time, and that those neurons fire a lot (both are required) : the network as whole has a lot of activity. If the FR is low, it means the opposite is happening : not many neurons fire, and the ones which fire don't do it at a high frequency. Intermediate values could be a bit harder to understand, as it could be because either few neurons fire a lot, or all neurons fire a little, but due to the AI nature of our networks and the homogeneity of the connectivity, it is more likely that all neurons will have, on average, the same activity. It is not obvious to understand everything about the variation of activity, but basics are easy : the firing rate linearly follow the global activity.

Vm is more difficult to interpret, because by definition the value of Vm for each neuron will go up until a threshold, after which it will go back at a resting value for a short time. This means we do not have a linear relationship between the activity and the Vm. A low average value of a single neuron Vm could either mean the neuron never receives much excitatory input and therefore never fires, which means the Vm value would only vary depending on the leaking nature of the neuron; or it could mean the opposite : the neuron fires a lot and the low value is due to the resting state, because as soon as the neuron is outside of it it quickly fires again, spending little time on higher Vm values.
In the end, the non linearity is not too much of a problem : it will be easy to identify if the network is on a low FR or a high one, and if it is intermediate, we will be able to see the VM slowly change, which mean we will still have changes correlated to the activity, but not as direct as the ones we have with the FR.

On top of that, while the FR only inform us on the activity, the Vm give information about the internal states of neurons, which is richer than only analyzing the activity. This is what we exploit in chapter 2, which allows us to determine the critical window to reverse the propagation that we talked about previously.

This is why we chose to use the Vm in chapter 2 and the FR in chapter 3.

Speaking of the Vm, in the introduction (section 2.3.2) we talked about how similar objects, as the Vm, would be represented with different equations in different models, and if it was alright to consider them

as the same object.

The two (main) different models we used in this work simulated the membrane potential in order to obtain spikes in different ways, but the next steps were identical : after a spike, an increment of conductance is sent to other connected neurons, which will increase or decrease their own Vm depending on the original neuron's population. The distribution of population and the connectivity are identical, therefore the only difference, in practice, will be the timing of the spikes generated by the evolution of the Vm.

While we could indeed observe differences in the neuron scale, they become harder to see at the network scale (although some can exist), and it is easy to obtain similar activity from networks made from different single neuron models. Therefore, it can appear that the two networks produce some similar dynamics and are mostly interchangeable, which we proved was not the case in this work.

This is actually very important, as different models can be use for various reasons without a deep understanding of their dynamics and only for their apparent functions or mechanism, making us often use models we do not really master to simulate a reality we do not really understand. In this thesis, we showed that despite those apparent similarities, studying the dynamics of the networks and how they react show different results.

We showed that AdEx networks have a lot of interesting behaviors : they show a bistability to propagate a paroxysmal input or to control it, a difference between Driven and Self-sustained systems, and it is possible to predict the response to a perturbation from its dissipation.

On the other hand, HH networks did not exhibit the drastic dichotomy we presented before, and did not appear to show an obvious link between its dissipation and response. While it means we could not generalize those finding, it does not invalidate them, and it also shows how different the behaviors of those networks can be, even when their activity seem similar.

Out of the differences we presented before, the most interesting one is related to the difference between Driven and Self-sustained networks that we explored in chapter 3 and chapter 4.

But first, even if there is no Self-sustained on chapter 2, we still have an interesting point to discuss regarding the Driven networks. Fig.17.a) shows that the influence of the drive is 10 times stronger than the one of the excitatory neurons, while they play the same role (same number of neurons, same gain of conductance, same number of connections). This mean the excitatory neurons are basically useless here, and the excitatory dynamics can be assimilated to a random behavior.

Of course, this could be different with different drives, and mainly different drive strengths, but it is still a fact that the drive will heavily influence the excitatory part of the network, and render it less deterministic.

While it could be an hint on why Driven and Self-sustained seem to differ in chaoticity, we have also seen

that Self-sustained with a drive have the same LE as normal Self-sustained, which means the presence of the noise and its disproportionate importance in the dynamic is probably not the only answer. It could be that in Self-sustained networks, the excitatory population is proportionally more important (as it is required to have a proper balance between the two populations of opposite effect). Or maybe the effect is same for Self-sustained and the excitatory population already does not play a major role in the dynamics, which explain why not much changes when we add another source of excitatory neurons, even if it is a purely random one.

Now, regarding the Driven and Self-sustained networks, we showed that their activity and dissipation seemed similar, but their were differences in their dynamics. The main result from chapter 3 was that Self-sustained networks had a FLE an order of magnitude higher than Driven one, and we showed how it could influence their dynamics. We confirmed that result in chapter 4, showing how the correlation between the FR and the dissipation was drastically different depending on the type of network, but not depending on the model. This is also our first result related to the difference between the two types of networks for HH, which makes us wonder why the differences were not apparent with the LE. More investigations will be needed to understand the differences between those two types of networks for other neuron models than AdEx, but we think it is likely such differences will indeed appear.

We claim here that, at least for AdEx network (but, as we said, potentially for other ones), the categorization between the two types of networks is a useful one, as it is rather easy to find if a network is Self-sustained or not (most of the time, it is created this way) and it leads to different dynamics.
The difference would be especially useful when related to perturbation, constant external drives or "connections" with other networks.
It is important to notice that we do not claim the dichotomy we propose is the only meaningful one, and that it could very well be that other apparently innocuous "details" of the model of a network (here, we are talking about all aspects of the network, not only the single neuron model) lead to important differences while still having an apparently similar activity.

We have talked about apparently big differences : a change of models of single neurons. Then, we talked about apparently smaller differences : a change of parameters within specific models. We want to talk about a final, apparently even smaller difference : randomness.
There is, of course, the noise of the external input that we talked about a lot in chapter 2, and that could greatly change the dynamic, sometimes leading to a pathological activity. But mostly, we want to talk about he randomness in the connectivity of the network. While it appears the network is homogeneous, because we do not create explicit differences in the topology of the connecting graph, there is in fact a distribution as shown in Fig.2. It is that heterogeneity that allowed us to group the neurons depending

on their inhibitory connections in chapter 2, which in turn allowed us to observe the behavior of the network in a more precise way than it would have been if we only considered the global activity (which would have been equivalent to thinking all neurons are the same).

That difference between the neurons is also shown in the distribution of their dissipation in Fig.33. Here, interestingly, it appears HH neurons follow some specific distributions while AdEx ones are gaussian, as one would normally predict.

But interestingly, it can actually be linked to another result we found and that we present in the Annex but did not have time to analyze further : the ranking. The idea was to create a measure that could link different temporal scale for the neurons : a local, "instantaneous" one and a global, averaged one over the simulation.

We ranked the neuron on those two scale, meaning we had ranks on the average firing rate of a neuron, and ranks on their local activity, which could allow us to see if neurons mostly always followed the same behavior or not. It is important to note that we are ranking the neurons, comparing them between each other, therefore this measure is a relative one.

We showed that AdEx neurons followed a monotonous relation between those two, meaning if a neuron was globally firing a lot, it was also mostly firing a lot locally (at least, more than the other neurons).

But the HH neurons were different : here, we observed a concave relationship : the neurons that were globally not firing a lot were indeed following the same trend locally, but then we would arrive to a maximum of local ranking and neurons that fired more globally would start to rank less than other locally.

This seem to be linked with the different distribution of HH neurons, showing that some neurons have a specific local activity when they sometimes fire a lot, and other time not as much. Of course, as we have AI activity, it is normal to have some irregularities. But over the course of the simulation, it would be expected for the differences to average out, which was not the case here.

This interesting and non trivial result could be a way to explain the differences we keep observing between HH and AdEx, but we did not have time to test those results enough to ensure their validity. Most of all, we did not have time to link the differences in the neurons to a difference in connectivity, although this is certainly a future work we will do in order to deepen our understanding of what the dissipation can tell us about neural networks.

**Conclusion**

In this thesis, we investigated the difference in apparently similar networks and found various interesting ones.

We showed that the AdEx networks show a lot of dichotomies such as the bistability on propagating or controlling an external input and a dynamical difference between different types of networks (Driven or Self-sustained), while HH networks did not show much differences and were rather continuous. This suggests the AdEx network is more variable, and probably has more dichotomies we did not looked for, meaning it is probably well suited to simulate phenomenon that can be strictly categorized. On the other hand, HH networks seem more continuous, without as much differences between different types of networks, which also means the choice of specific parameters seem less important and the effect on HH networks will be more robust, with less variabilities.

We showed that Driven and Self-sustained networks had, on top of the obvious difference of the external drive and the way we construct them, important differences in their dynamics. On top of that, we showed in chapter 3 that those differences remained even when both had a similar external drive. This could lead to accidentally use a Self-sustained network instead of a Driven one and to find results that would not generalized for actual Driven networks.
We showed that such difference was mostly present for AdEx networks, but that there was also indications it existed for HH ones (and probably others), and we conclude that this dichotomy should be pushed in the analysis of various neural networks.

We also showed that the connectivity could have an influence on the global behavior, and that the conception of random neural networks as homogeneous could be a problematic ones considering the important variabilities that exist. It would be better to talk about uncontrolled heterogeneity if the goal is to differentiate it from cases with specific topology, for example.

We found indications on how AdEx networks behave, from how it responded to external inputs to how it could change through the Driven/Self-sustained dichotomy, which deepen our understanding of those networks.

In this thesis, we introduced a tool from dynamical systems which is rarely used in the computational neuroscience field : the dissipation.
We claim that this tool is a powerful one that allows us to explore the dynamics of high dimensional networks in a different and easier way than the Lyapunov Exponent analysis, and that is highly informative, enabling us to link different scales of analysis, from single neurons to the whole network, or unravelling systematic differences between Driven and Self-sustained in both AdEx and HH networks, something that we could not obtain with LE.
The use of dissipation to study neural network should therefore continue, to understand properly everything it could tell us and what are the limits of that method.

Finally, the main results we want to emphasize on is the need for a more systematic analysis of the neural

networks we use. Those objects made from single neurons model that we mostly understand let complex dynamics emerge that we are still far to completely control, and it is likely that a lot of apparently similar activity have actually a crucially different dynamics which would give us different answers on important questions we ask. Understanding those networks dynamics better, understanding their specificities, their differences or similarities, and why they exist, would in turn help us use those tool more accurately and, in the end, understand the brain better.

# Annex

In this Annex, we will present more results that were either not conclusive enough, that required more time to properly analyse, or that were simply not important enough to be in the core of this thesis.

First, we will present additional work from chapter 3, showing more analysis on the differences between the variables used to reconstruct the average system for the Driven and the Self-sustained network.
We also tried to apply Takens theorem to reconstruct the network from only one of the variable, and we show the results.

Then, we will show additional work from 12. Here, we will start by presenting more work on the link between temporal evolution and the dissipation that we briefly introduced in 15.2. A special interest should be taken for the "ranking" results, although they need to be investigated more.
Then, we will observe the link between the membrane potential Vm and the dissipation at different scales.

Finally, we will provide the main codes that we used in this thesis : the simulation of neural networks (for AdEx) and the way to compute the FLE from time series.

# 17 Complementary results from chapter 3

As mentioned before, we will start by the complementary results from chapter 3. Those results aim to get a better understanding of the system we use to compute the FLE, but the difference between driven and Self-sustained system, while apparent, would require a deeper analysis that we did not have the time to do.

## 17.1 Behavior of the average variables

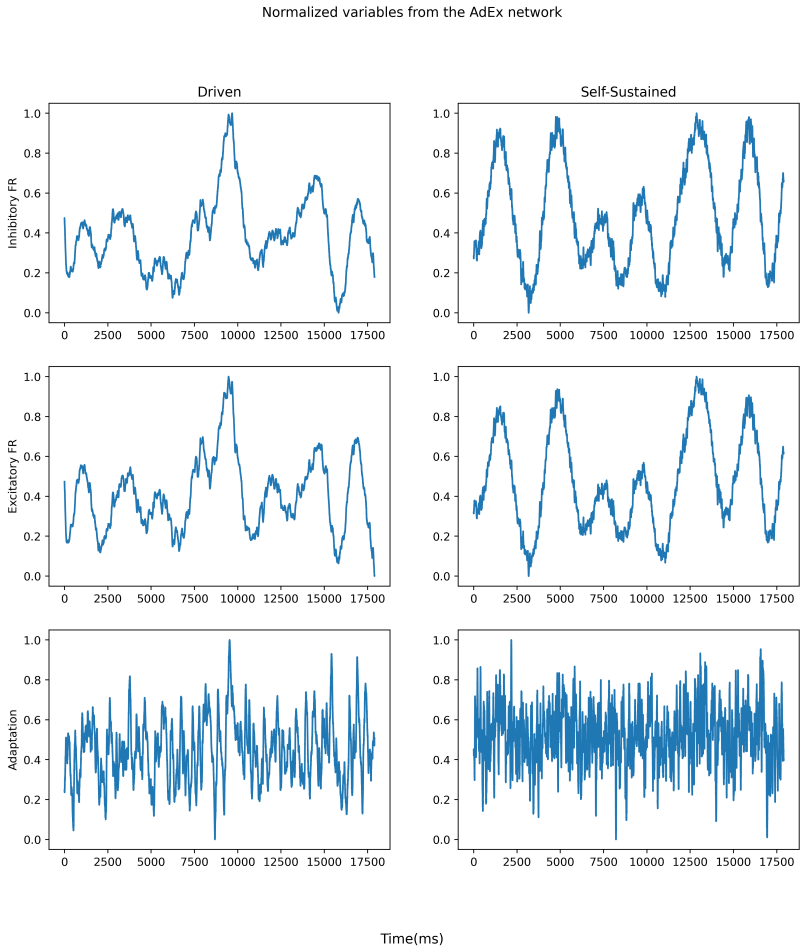


Figure 46: The three different variables in lines, respectively : the firing rates (FR) of the inhibitory and excitatory populations and the adaptation variable, for both driven (left column) and self-sustained (right column) systems. All variables were normalized to be compared and to apply the algorithms.

In Fig.46 we can see the averaged, normalized variables look similar between driven and self-sustained, although the driven ones are smoother versions of the self-sustained ones, which is already an interesting difference to point out. The firing rates also look very similar between them, while looking different from the adaptation variable. While both variables are related to the spikes in the system, the adaptation also has its own intrinsic dynamics, which explains why it is less smooth that the firing rates.

We also studied the same thing for HH networks, and we can see in Fig.47 that the firing rates seem less rough than with the AdEx networks, and also that the apparent distinction between the two of them seem to have vanished. Then the figure shows that the rest of the variables are very noisy, with different average values. While there seem to be some differences between Driven and Self-sustained networks, no obvious pattern emerges.

In both AdEx and HH networks, it appears that there is a difference within the values of the variables we consider when we compare them between Driven and Self-sustained networks (especially for AdEx networks), although the difference seems fairly weak, which is why we did not focus on that part.

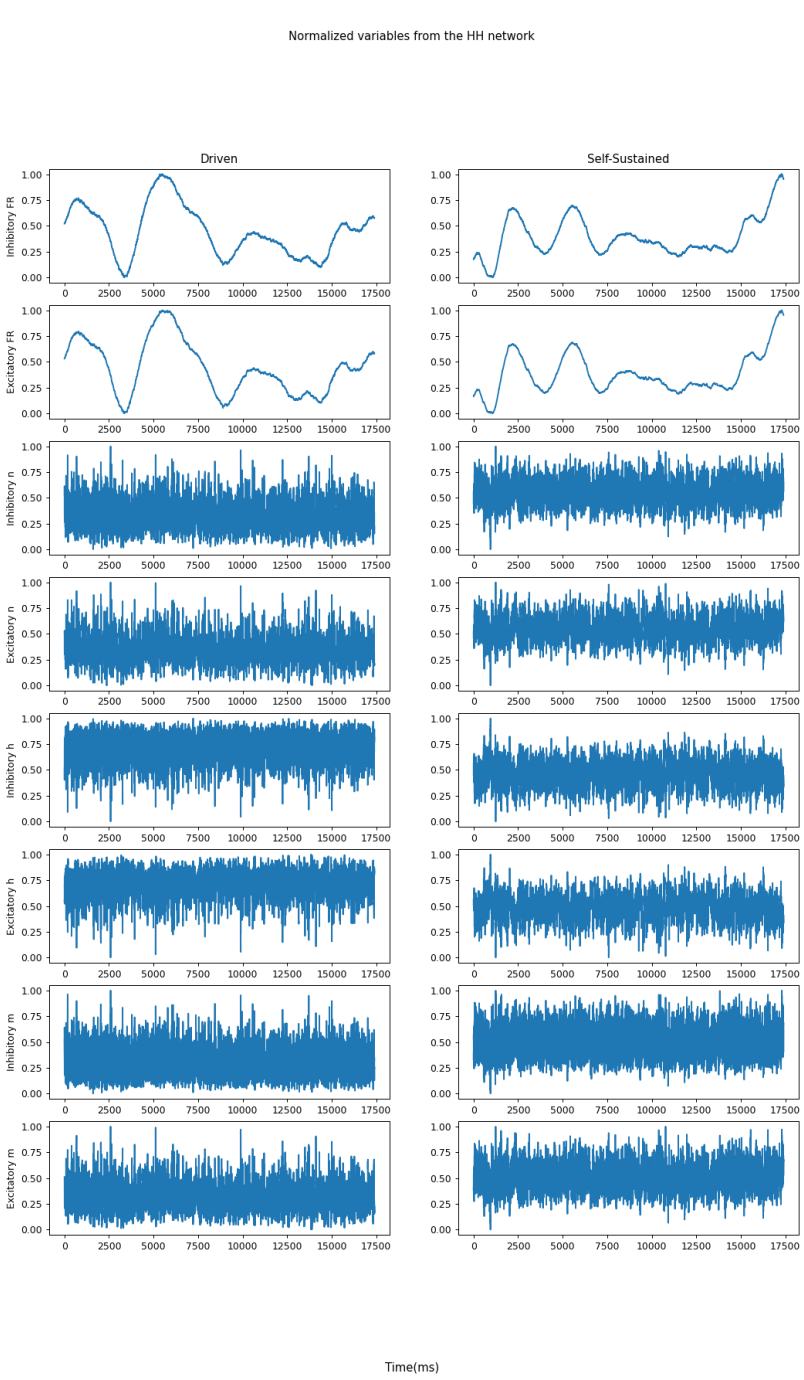


Figure 47: Representation of the HH network.
A)-B) : Raster plot and average firing rates of a driven (A) and self-sustained (B) HH network, with inhibitory neurons in red and excitatory neurons in green (same for the averages).
C)-E) : Normalized variables of interest as a function of time. Firing rates of the inhibitory and excitatory populations for the driven (C) and self-sustained (D) network. E) Represent the rest of the normalized value, in order : n,h and m, each times with the inhibitory and excitatory population. The left column represent the variables from the driven network, while the right column is the variables from the self-sustained network.

## 17.2 Power spectra and correlations

To further compare the two types of networks, we computed the power spectrum and the cross correlations, as shown in Fig.48.

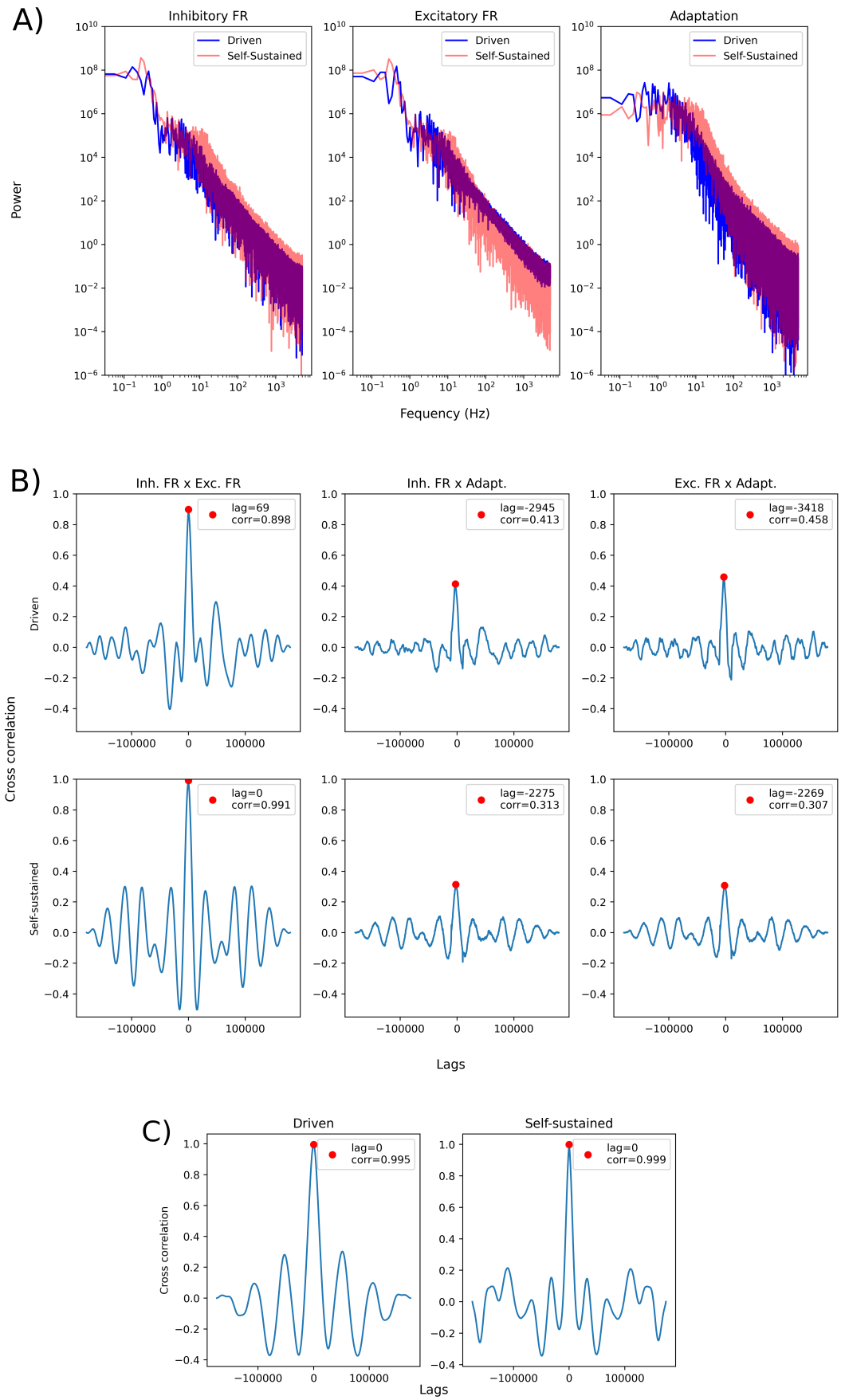


Figure 48: Analysis of the three main variables. A)-B) for AdEx : the 2 Firing Rates (inhibitory and excitatory) and the Adaptation.
A) Power spectrum of the three variables.
B) Cross correlations between the 3 variables (as a function of some lag). The red point indicate the global maximum correlation, with its value and the associated lag.
C) is for HH : the cross correlation of the two firing rates, for driven and self-sustained networks. Same as B).

As we could have seen from Fig.46, Fig.48.A) shows a vertical shift between the Driven and Self-Sustained networks power spectrum, indicating a higher power of the later, but the difference is not that big and the excitatory neurons part makes it harder to completely understand what is happening here.

Then, parts B) and C) show the cross correlation between the variables of interest (all of them for AdEx, only the two FR for HH) for both Driven and Self-sustained networks. The main thing we can see is that the FR are always very correlated in all cases, more than the other variables.
There is (again) not much difference between the two types of networks for HH, but a small difference appear for AdEx : Driven networks have a slight lag for their maximum correlation compare to Self-sustained ones, and are also less correlated. The link between the two FR is not as strong for Driven networks than for Self-sustained ones.
This could be due to the presence of the drive, and it might explain the difference in LE. At least, it would be a good path to investigate.

## 17.3 Takens reconstruction

We also used the so-called Takens reconstruction (Takens, 1981) to calculate the FLE, following the procedure described in (Wolf et al., 1985), as shown in Table 15.

| Driven | Inh. FR | Exc. FR | Adapt. |
|---|---|---|---|
| Dt = 10 | 0.028/0.017 | 0.014/0.069 | 0.080/0.038 |
| DT = 20 | 0.024/0.021 | 0.015/0.023 | 0.092/0.033 |
| Self-sustained | | | |
| Dt = 10 | 0.026/0.031 | 0.008/0.023 | 0.109/0.106 |
| DT = 20 | 0.018/0.031 | 0.017/0.017 | 0.105/0.107 |

Table 15: FLE of the networks reconstructing from one dimension (the Excitatory and Inhibitory Firing Rates, and the Adaptation, in columns) with the Takes reconstruction method. The first part is for a driven network, and the second part for a self-sustained one. The lines represent 2 different time steps used in the algorithm, to ensure some robustness. Finally, each condition was repeated with a different noise realization (including both the drive and the connections in the network).

Here, interestingly, we see that we do not have the same results as those shown in chapter 3 : there is no apparent differences between Driven and Self-sustained, or at least noting as big as the order of magnitude we had previously. Interestingly, we can see reconstructing the network from different dimensions lead to different FLE, so the global geometry of the network is not reproduce correctly. This means it was most likely not possible to apply Takens theorem here.

A possible reason as to why it was not possible might be that, while the 2 spike timings and the adaptation are definitely causally linked, it is far from obvious that their average value would follow the same trend. Most likely, they each still carry relevant information about themselves, but fail to carry the one from the other variables : it was lost with the averaging. Interestingly, we can see the reconstruction from the Adaptation seem to follow the trend we found in chapter 3, and the adaptation is an intrinsic dimension

from the network contrary to the FR which is an external observable (similar to what we discussed in the Global discussion between FR and Vm).
It could also be interesting to dig in that direction in order to better understand the average system we produced.

# 18 Complementary results from chapter 4

## 18.1 Temporal evolution and Dissipation

As we said in chapter 4, dissipation is a rate of contraction in the phase space. Therefore, we will focus here on an important part : the link between dissipation and temporal evolution of the system.

### 18.1.1 Recovery

First, we aimed to check the recovery time of the network to a perturbation, similarly to what we did in the Introduction chapter for the Lorenz system. We will use driven networks here.

To do so, a square input of 2 lengths (100ms and 500ms) was injected in the Poisson noise (adding it to the drive). The input can have two different amplitude (2 and 10 Hz). The driven networks have drives of 2Hz and 10Hz. All firing rates have been binarized in bins of 5ms to smooth things up.

Two FRs (average of the whole network) that both last 500ms were used, which forms 2 datasets. The first one is taken at some point way before the perturbation, the second one starts at the offset of the perturbation and is repeated with a sliding window every bin of 5ms until the end of the simulation. Each comparison gives a Z score.
Therefore there is an estimation of how different each sequence of the sliding window is from a random sequence before the perturbation. As the sequence is long enough, the system "recovery" can be defined as when the absolute Z score goes below a given value, that was fixed at 2.56, corresponding to a p-value of 1%.

**First, let's focus on AdEx.** We represent the neuron z-score in Fig.49.

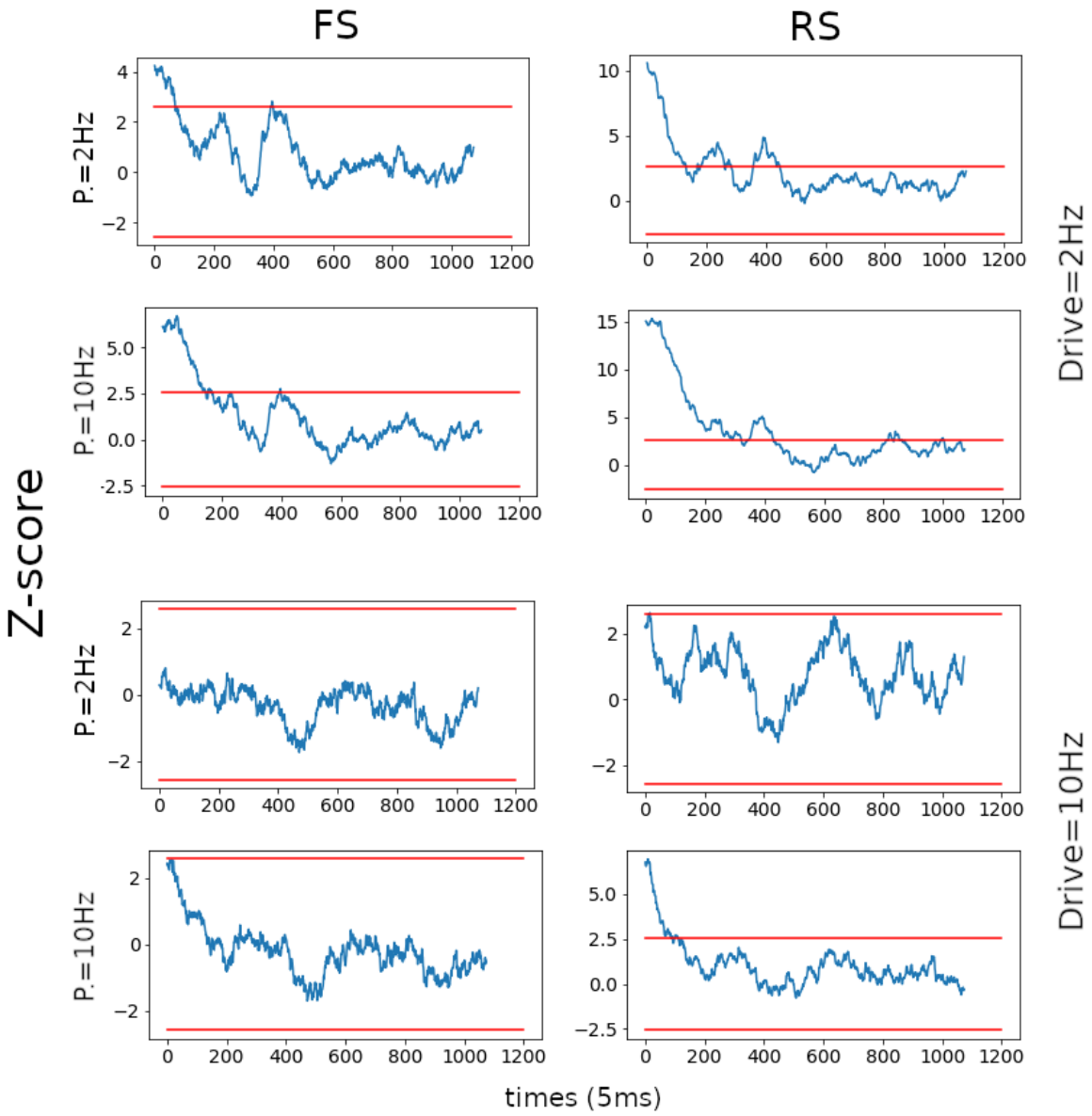


Figure 49: Evolution of the Z-score as a function of time while the networks "recovers". We represent the inhibitory (left) and excitatory (right) populations for an AdEx network, with various perturbation as lines (P.=), and various drives, also in line (Drive=).

For inhibitory neurons, only the 2Hz drive takes some time to arrive below the line, the other drive is already below it just after the end. We can see the strength of the perturbation has some effect, as it takes more time to go below the line for 10Hz perturbation than for 2Hz one.

Excitatory neurons appear wilder, with a big amplitude. Apart from that, the same effect is observes (although it takes longer to recover), and there is also some recovery time for the 10Hz drive big perturbation (shorter than from the 2Hz drive though).

**Then, for the HH** (Fig.50).

It never goes above the line. It already dissipated so much it doesn't change after the perturbation, this makes sense considering the dissipation of HH networks is 20 times stronger than the AdEx ones.

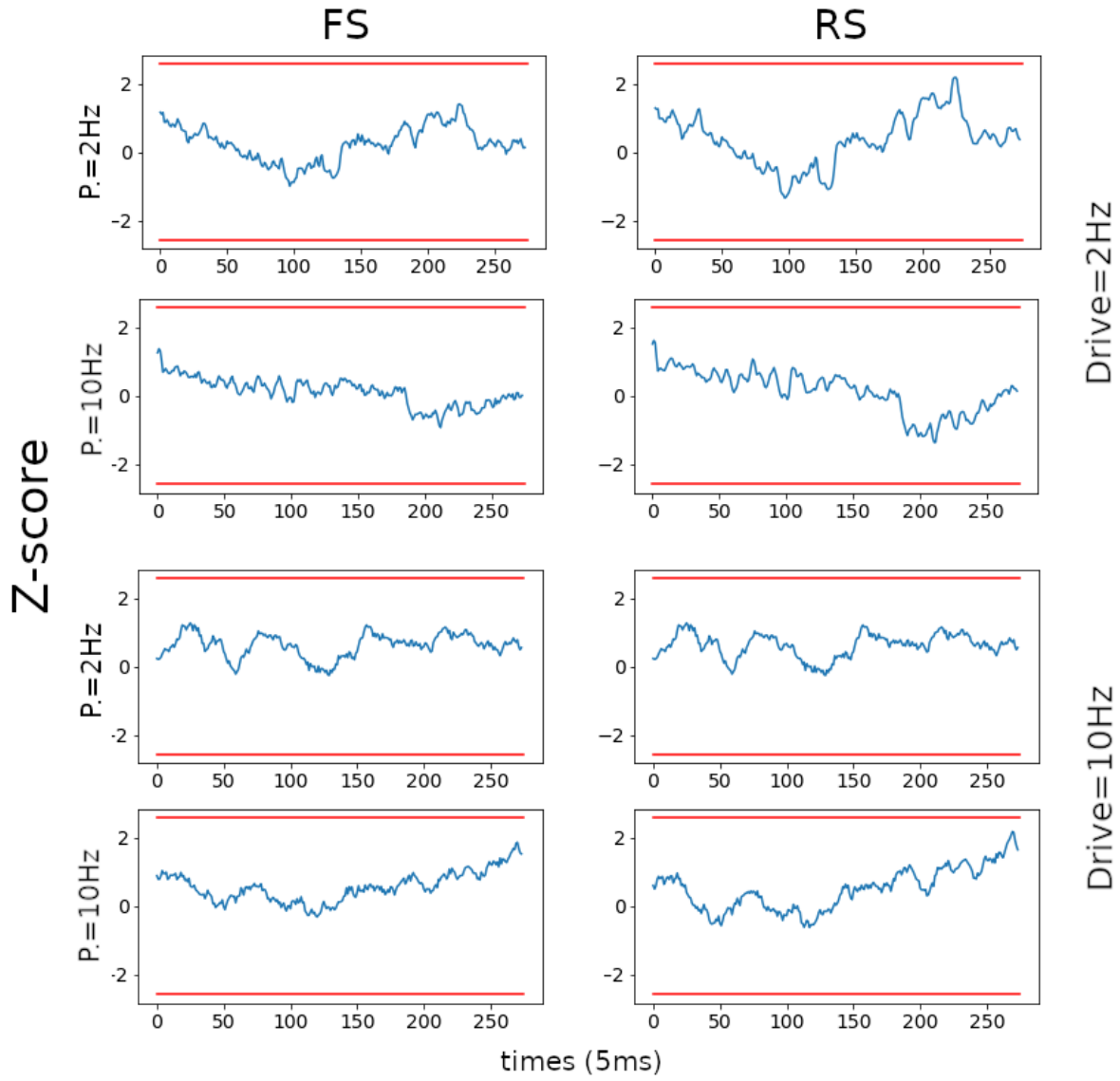


Figure 50: Evolution of the Z-score as a function of time while the networks "recovers". We represent the inhibitory (left) and excitatory (right) populations for an HH network, with various perturbation as lines (P.=), and various drives, also in line (Drive=).

It appears that HH network go back quickly (or never quit) there "normal" trajectories compared to AdEx, which makes sense as their dissipation is way higher.

On the other hand, it's not a simple linear relationship, so it's hard to say more. Maybe there is a bifurcation that "stops the change" when the dissipation is strong enough, or maybe no such conclusion can be taken with only two models and it's just that specific things regarding the dissipation can happen to specific models, but no generalization can be made.

It would be interesting to try different source of noise and maybe to have a more precise cut-off so that we can see a difference for HH and actually quantify the difference between the two model.
Here, only one Driven network was used (with different noise realization that gave the same results and are not shown here), it would be interesting to test other driven networks and compare them to Self-sustained networks.

### 18.1.2 Ranking

Here we provided a different analysis through time, trying to link local and global time scales. We wanted to know whether a neuron that dissipate on average more than other would also tend to dissipate at each time bins more than others, or if there were strong local variations. To put it in another way, we wanted to know if the "dissipative role" of a neuron (dissipate more compare to the other neurons, or dissipate less) was homogeneous through time.

To do so, we computed the average dissipation through the simulation for each neurons and ranked all the neurons, obtaining the global rank.
Then, for each bin of 5ms (and 20ms, but we do not show those results here as they were redundant), we computed the average dissipation and ranked locally the neurons. We therefore had a ranking for each bin, and we took their average to have the average of the local rank.
Plotting them against each other, we expected to observe a sigmoid (border condition would make it difficult for the average local rank to have extreme values).

Fig.51 shows the result for AdEx networks, representing a driven (up) and a self-sustained (down) network. For the driven networks, the result is as expected although it was not exactly a sigmoid and the border looked a bit weird : the low values of the local average are actually accessible, but the high ones are not). Self-sustained networks has the same result for excitatory neurons, and something more symmetric for inhibitory ones (still not a sigmoid).
It is not exactly as expected, and the differences could be worth investigating, but nothing is too surprising here.

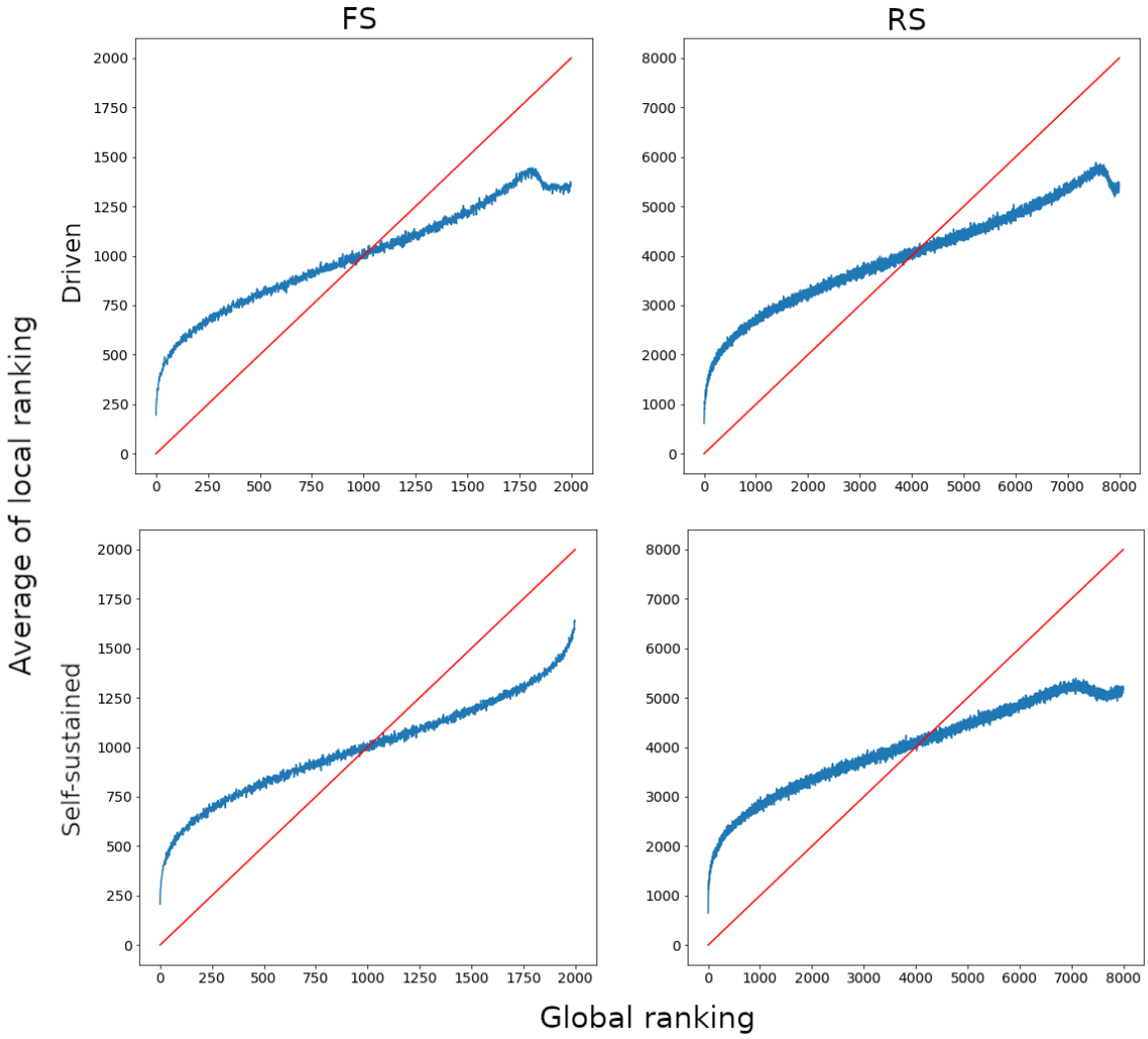


Figure 51: Average local ranking as a function of global ranking for all neurons in a driven (up) and self-sustained (down) AdEx networks. We separated inhibitory neurons (left) and excitatory ones (right)

The really non trivial result comes from HH in Fig.52, corresponding to a driven (up) and self-sustained (down) networks. The excitatory driven is mostly similar to before (although the very low general rank are weird), but for the other ones, we have at some point an opposite effect : when the global rank goes up, the local rank goes down.

Those results suggest the neurons can have some short time when they dissipate a lot (enough to be ranked high on average) but dissipate less than the other at the majority of bins. It means they have a different "role" at different time scales.

**To conclude** Recovery gave us some new clarifications on how the dissipation affect the networks, and some more differences between AdEx and HH.

But the true differences arise for the ranking representation, that also allows us to link different time scales.It would be interesting to learn more by checking the connectivity of those specific neurons, to see

if the heterogeneity that comes from the random connectivity of the network does indeed make neurons having specific "roles" in the network. Then, it would be interesting to understand the functional role of those neurons.

Please note that those results should be confirmed with other simulations and are therefore only indications for now.

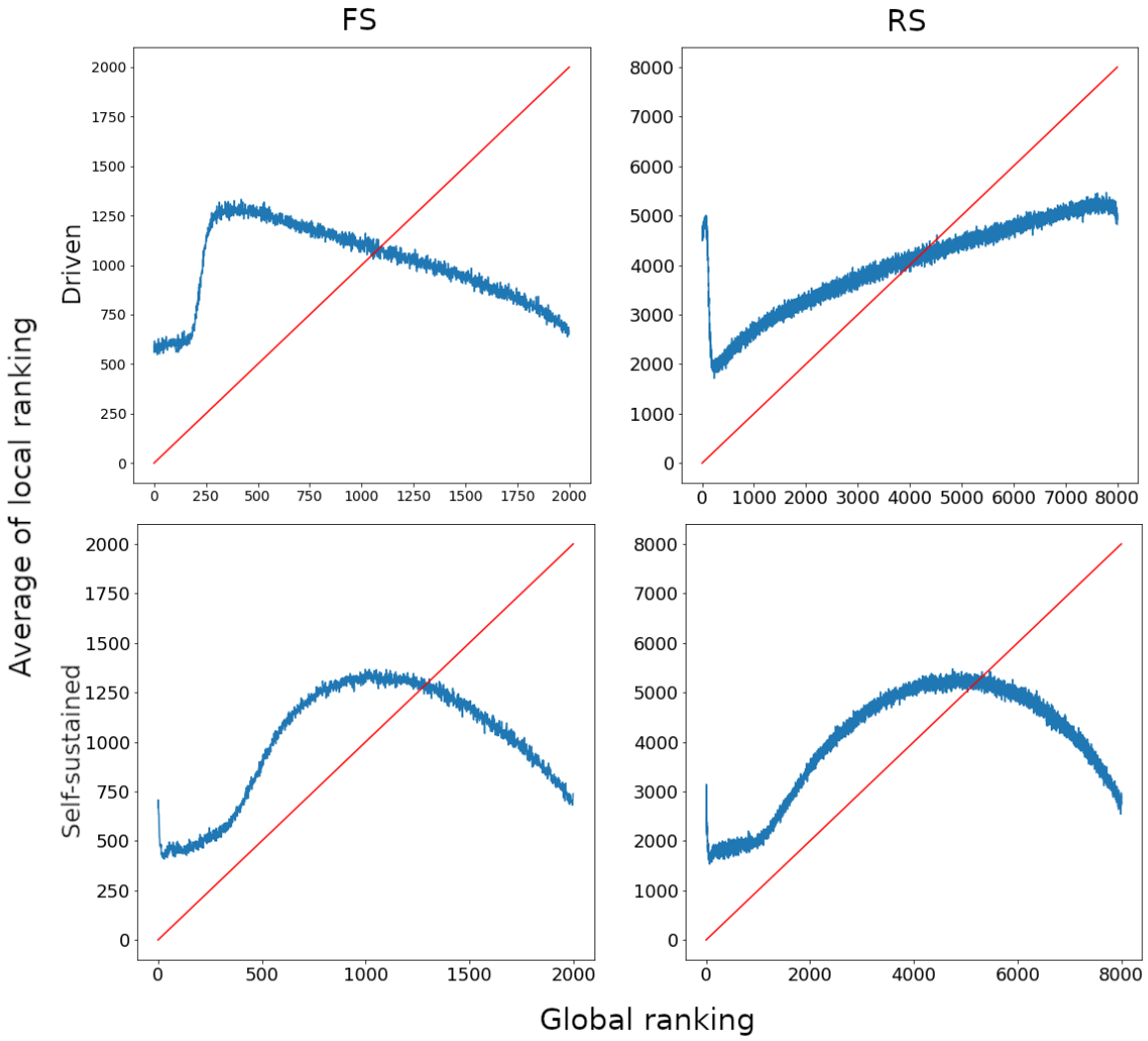


Figure 52: Average local ranking as a function of global ranking for all neurons in a driven (up) and self-sustained (down) HH networks. We separated inhibitory neurons (left) and excitatory ones (right)

## 18.2 Vm and dissipation

After looking at the temporal evolution of the network, we want to see the links between the dissipation and some important inner variable of the model : the membrane potential.

Here we stop analysing networks as a whole and focus on a few specific neurons (5% of them, chosen randomly). Each neurons will give us their Vm and dissipation at each time step of 1ms (which is way we couldn't take all of them : there would be way too much data). The simulation is done for AdEx and

HH, for driven (2,5,10 and 20 Hz) and self-sustained, with 2 realizations.

### 18.2.1 Whole network

First, following what we did on chapter 4 for the FR, we analyse the correlation between Vm and the dissipation for each neuron during the simulation. We then look at the distribution of the correlation. For driven networks, the columns for driven are the drive, then two lines represent inhibitory neurons in two different noise realization, then the 2 next are for excitatory ones, as before. For Self-sustained networks, only two noise realizations are shown, which again a separation between excitatory and inhibitory neurons.

Figures 53 and 54 represent AdEx networks (Driven and then Self-sustained), while Figures 55 and 56 represent HH networks (same).

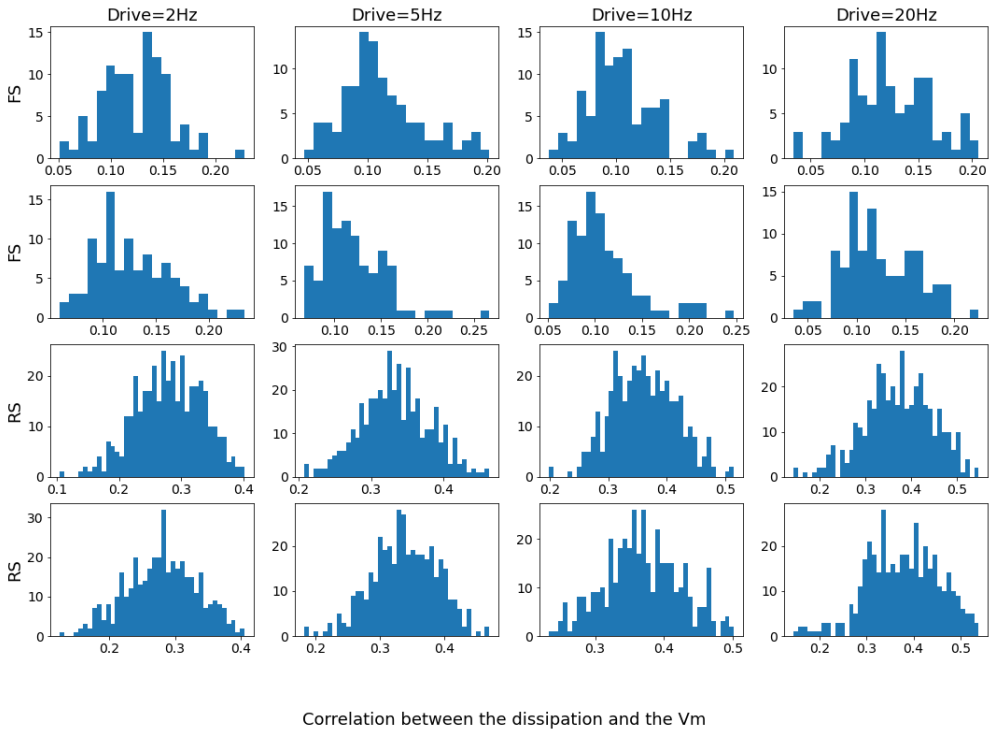


Figure 53: Distribution of the correlations between Vm and dissipation for each neuron for a whole simulation, for an AdEx network with various drives

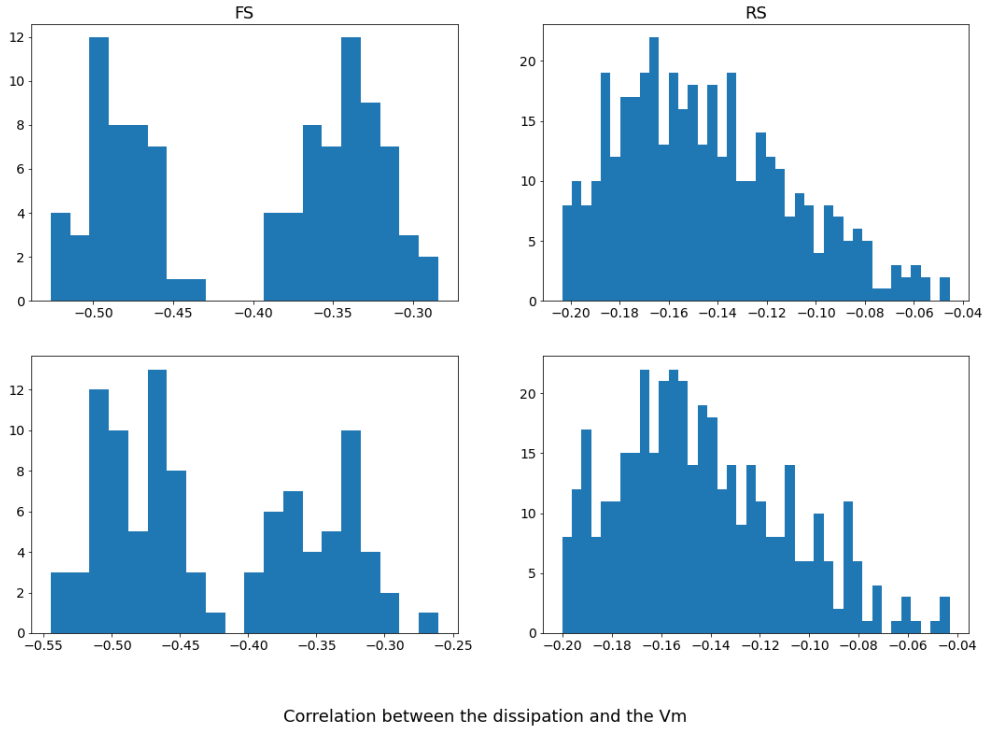


Figure 54: Distribution of the correlations between Vm and dissipation for each neuron for a whole simulation, for a self-sustained AdEx network

First, we can see that the Driven network in Fig.53 always have positive correlation, and the distribution is more or less Gaussian. It's fairly weak for FS (0.15, 0.05 wide) and a higher for RS (0.35, 0.1 wide). There is no obvious effect of the drive.

The Self-sustained network in Fig.53 always shows a negative correlation.

That difference echoes with what we found in section 15.4, although the two measurement do not represent the same thing. As we said in the discussion, Vm and FR both measure the activity and are effective at different scales. Here, at the scale of the neurons, it appears the correlation is deeper than for the FR at the scale of the network.

The FS have this time a stronger (anti) correlation than RS. RS is Gaussian-like with a tail toward 0 (average around $-0.15$, 0.05 wide without the tail). FS on the other hand is a binomial around $-0.48$ and $-0.35$ and about 0.05 wide each.

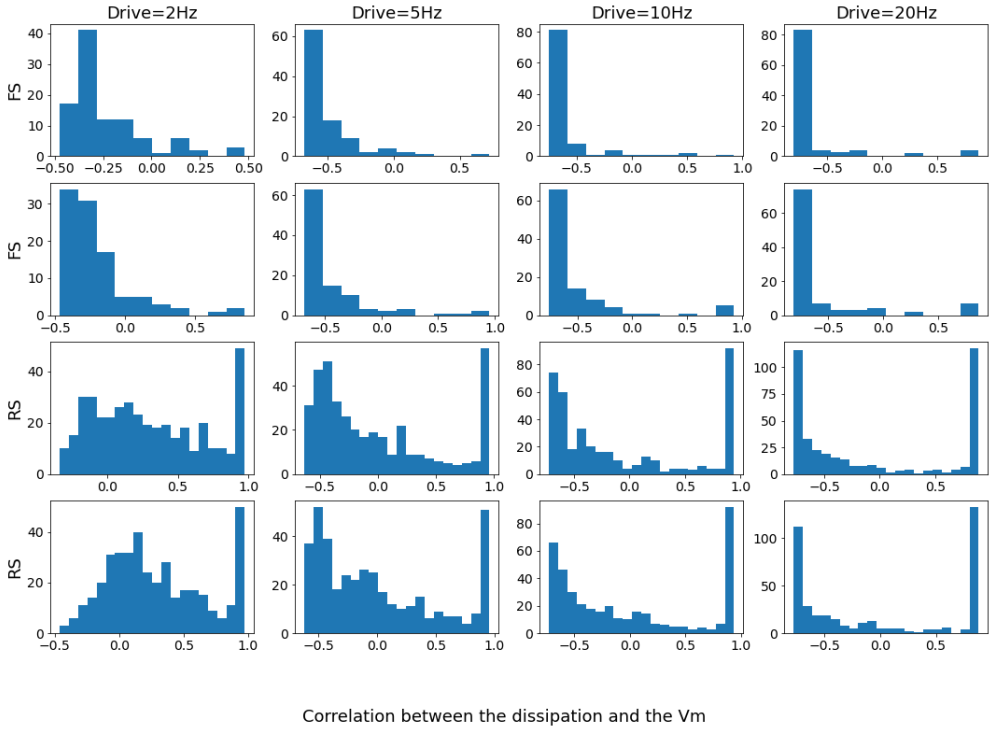


Figure 55: Distribution of the correlations between Vm and dissipation for each neuron for a whole simulation, for an HH network with various drives

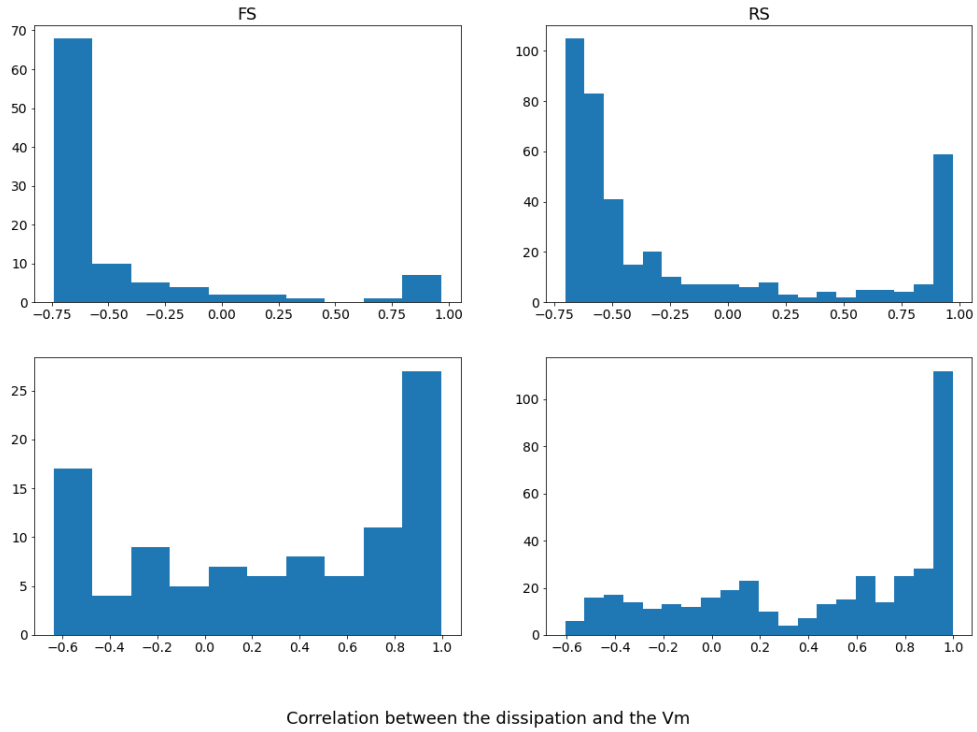


Figure 56: Distribution of the correlations between Vm and dissipation for each neuron for a whole simulation, for a self-sustained HH network

HH is, yet again, different, as the Driven(Fig.55) and Self-sustained (Fig.56) go for very anti-correlated to very correlated, being fairly polarized (with few values in between).

For the Driven network, the FS neurons show a peak in high anti-correlation (around $-0.5$) then few values a bit everywhere. The drive seem to be correlated with a bit more values in the very correlated region, but it's not impressive.
The RS on the other hand has a bunch of different anti-correlation and then a peak of high correlation. There is a clear effect of the drive : a weak drive has uniform values (between $-0.25$ until $0.75$) and then a peak, and his way closer to the positive correlation. Higher drives polarize more and more, making the values closer to $|0.8 - 0.9|$.

The self-sustained varies depending on the realization, but we can see it's also polarized between $-0.7$ and $0.9$, although a realization has more in between.

In general, it would be useful to compare those results with more Driven and Self-sustained networks to see if a trend seem to emerge. As with many other results we have previously, it appears AdEx networks have a lot of possible dichotomy while HH networks seem more continuous with way less change between different conditions.

### 18.2.2 Specific neuron

Then, we analyses neurons directly by checking Vm as a function of the Dissipation. We represent 9 neurons to develop a feeling of what they do.

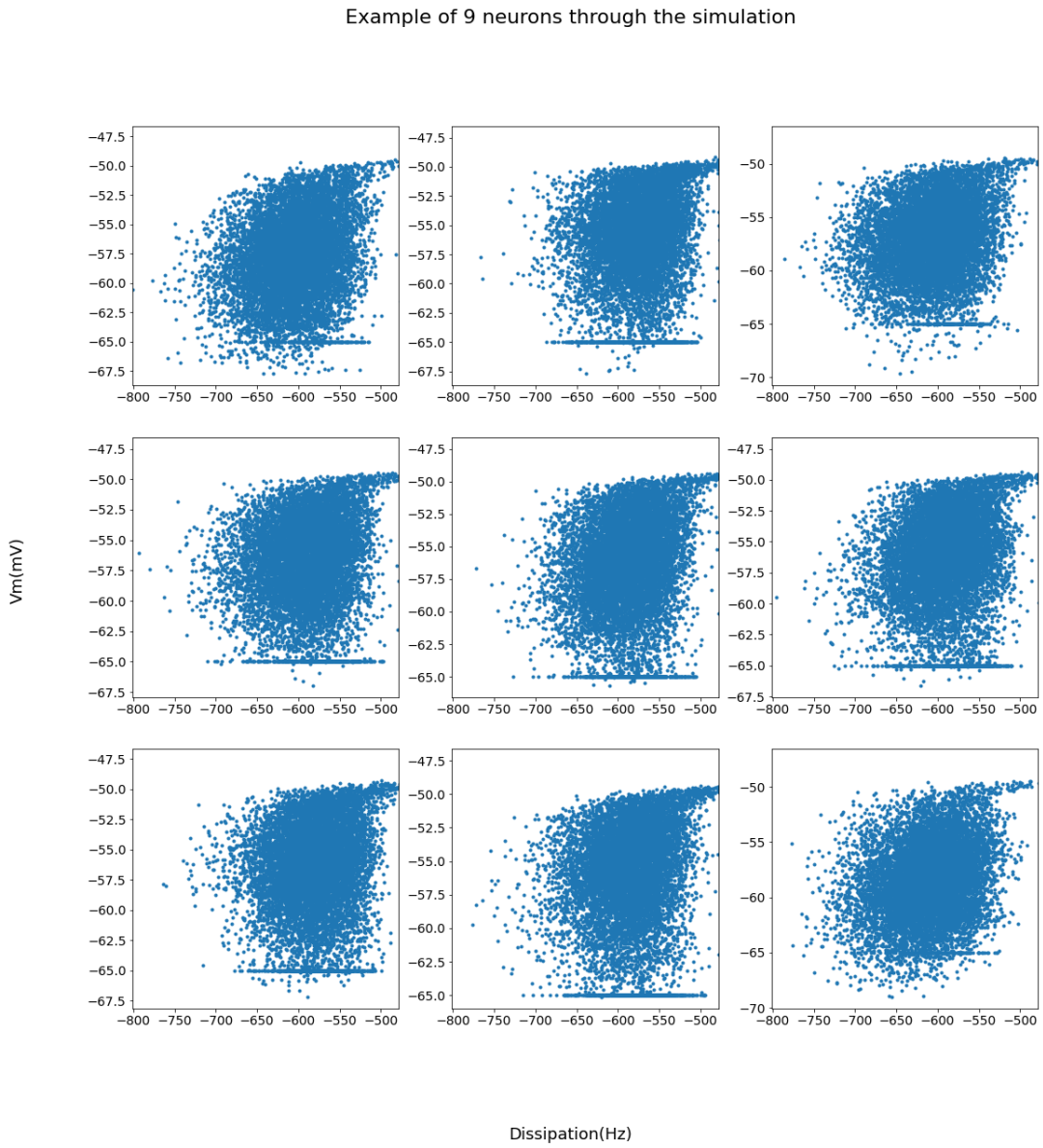


Figure 57: Example of AdEx driven neurons with a drive of 2Hz from the FS population

We can see two things for AdEx driven(Fig.57 (similar results for RS neurons are not shown) : a kind of tail on the part with higher dissipation and Vm, and the line of -65mV corresponding to the rest potential after a spike. The form does seem more correlated for RS than FS. The tail part correspond to the potential just before the spike, and gives access to the weaker dissipation (closer to 0, as expected because of the exponential term in the equation).

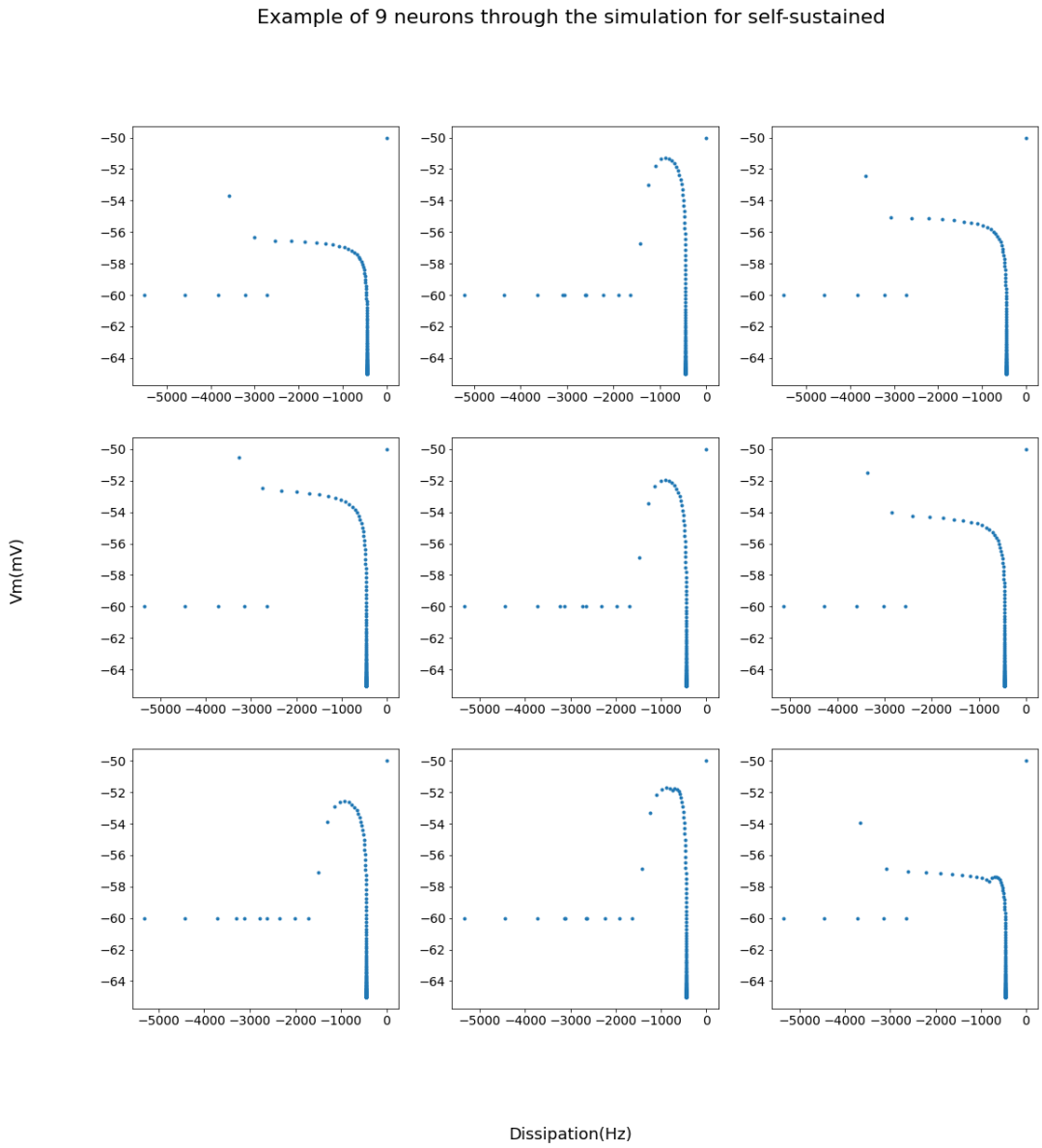


Figure 58: Example of AdEx sustained neurons from the FS population

The self-sustained part is less trivial, as can be seen in Fig.58 : there are clearly two parts, one almost horizontal line, and one that will hit a wall at a given value of dissipation : -450Hz.// Finding why could lead to interesting discoveries and a better understanding of the differences between driven and self-sustained systems.

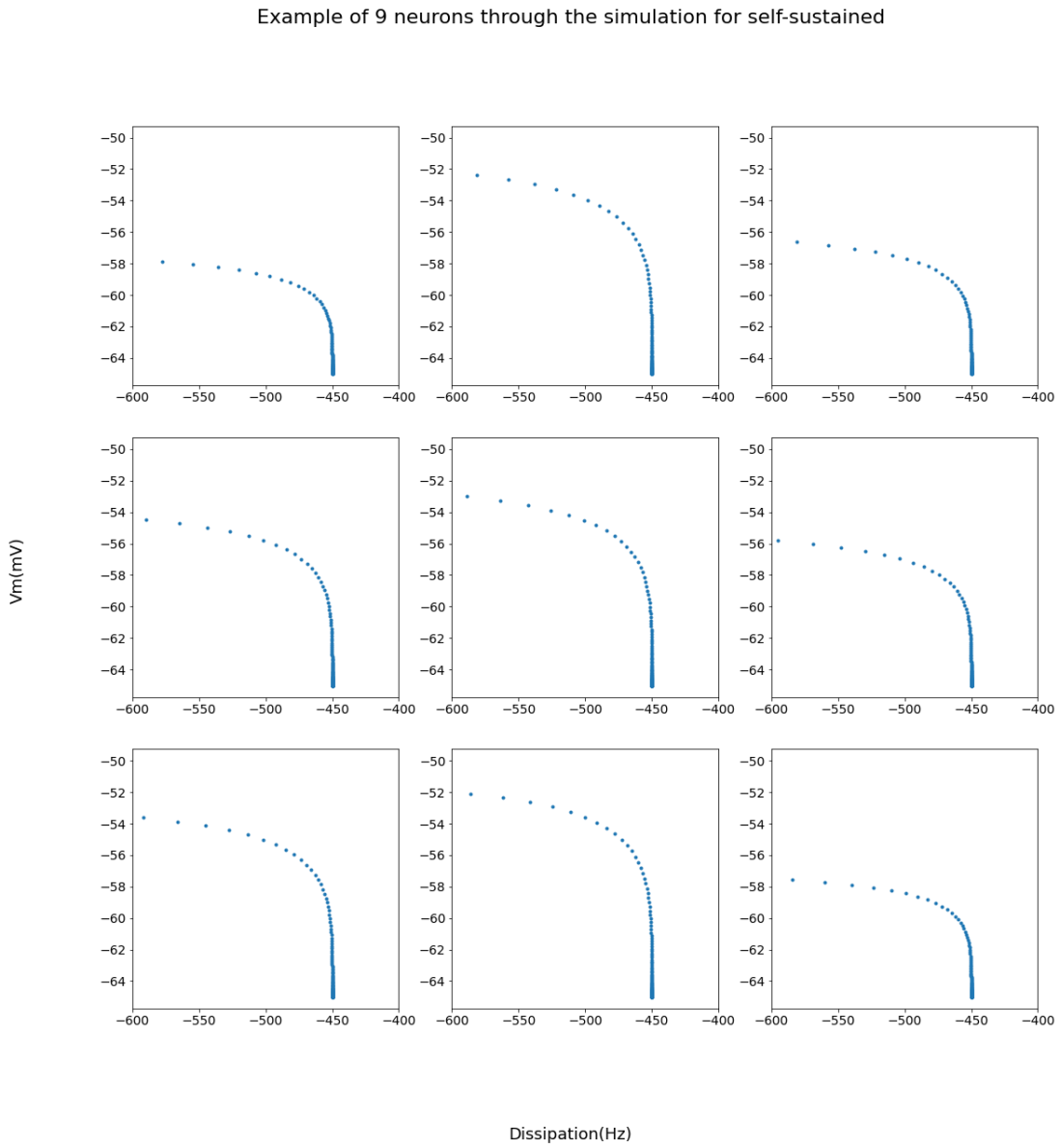


Figure 59: Example of AdEx sustained neurons from the FS population, zooming on the lower part of the dissipation

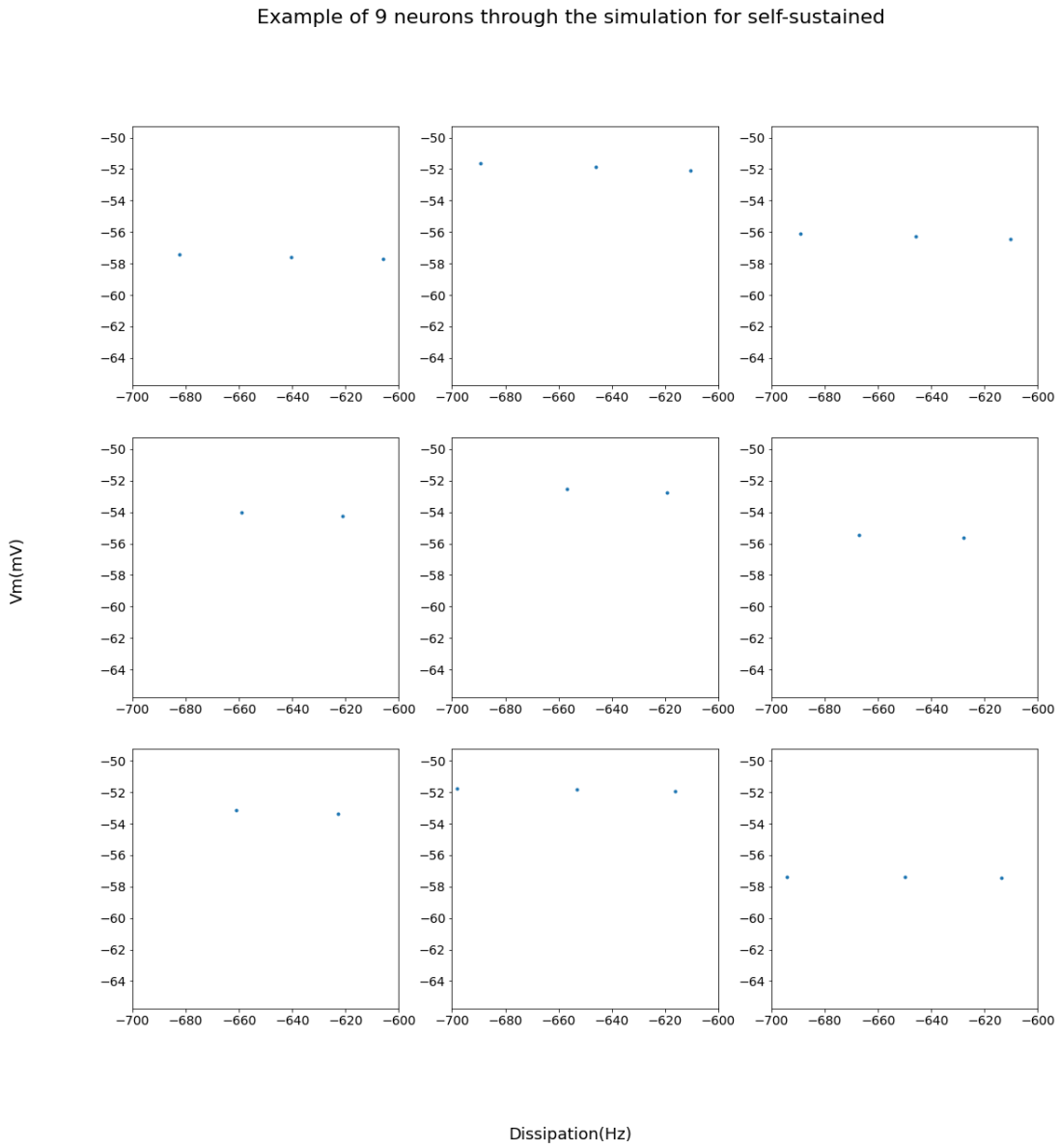


Figure 60: Example of AdEx sustained neurons from the FS population, zooming on the higher part of the dissipation

Actually and non trivially, almost half of the points are contained in two or three different spots of similar Vm and different dissipation Fig.60 (here only the FS population is represented, but it's similar for RS), while the other part is more "normal" Fig.59 which means self-sustained AdEx is locked in specific states most of the time.

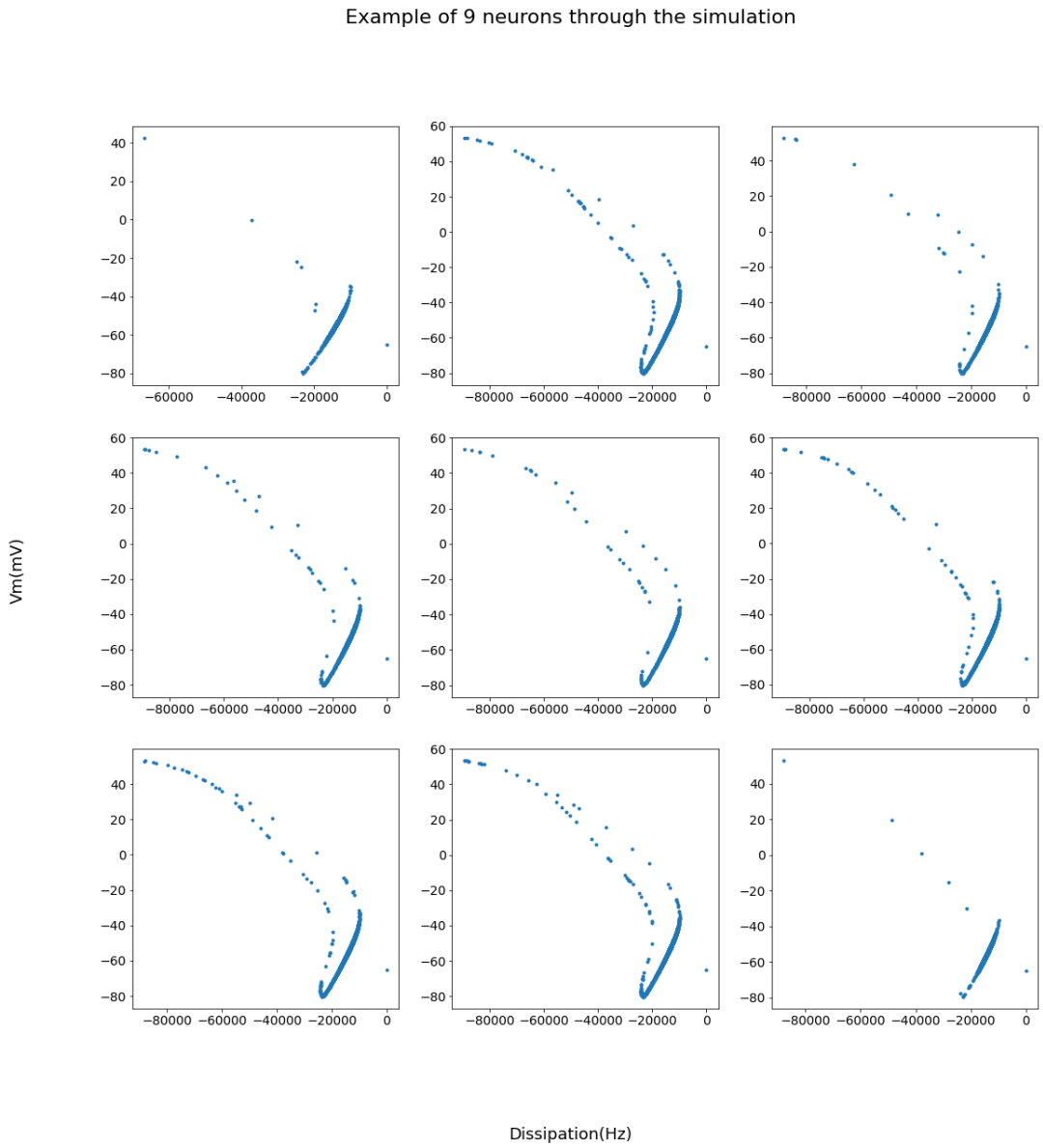


Figure 61: Example of HH driven neurons with a drive of 2Hz from the FS population

The HH (Fig.61 gives a figure with two clear behavior : one anti correlated for strong Dissipation high values of Vm (often positive) and one correlated for weaker Dissipation and lower values of Vm. The same result was given for self-sustained networks. We can see where the two kind of correlations come from, but it's hard to know why, for some neurons, the correlation turns positive and for other negative. Maybe, as for the self-sustained of AdEx, few positions are actually accessible in the phase space, and there are a lot of them in one our the other situation that is not really visible here.

### 18.2.3 Correcting the HH network into 2 subsets

We hypothesize that those two region of dissipation were giving the different correlation, and that it might have been an issue with the Pearson correlation formula that gives more strength to few very high values.

We therefore made 2 subsets : one for the strongest 5% dissipation and the corresponding Vm (that we expect to have a negative correlation) and the other for the rest (expecting a positive correlation).//

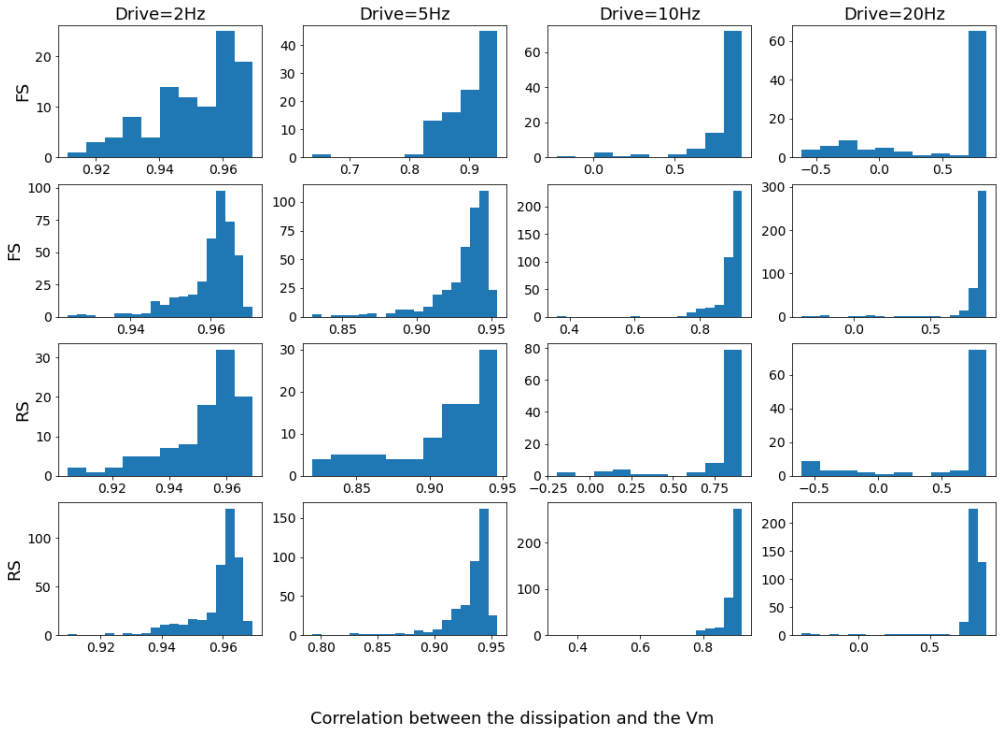


Figure 62: Distribution of the correlations between Vm and dissipation for each neuron from the previously defined higher subset for a whole simulation, for an HH network with various drives

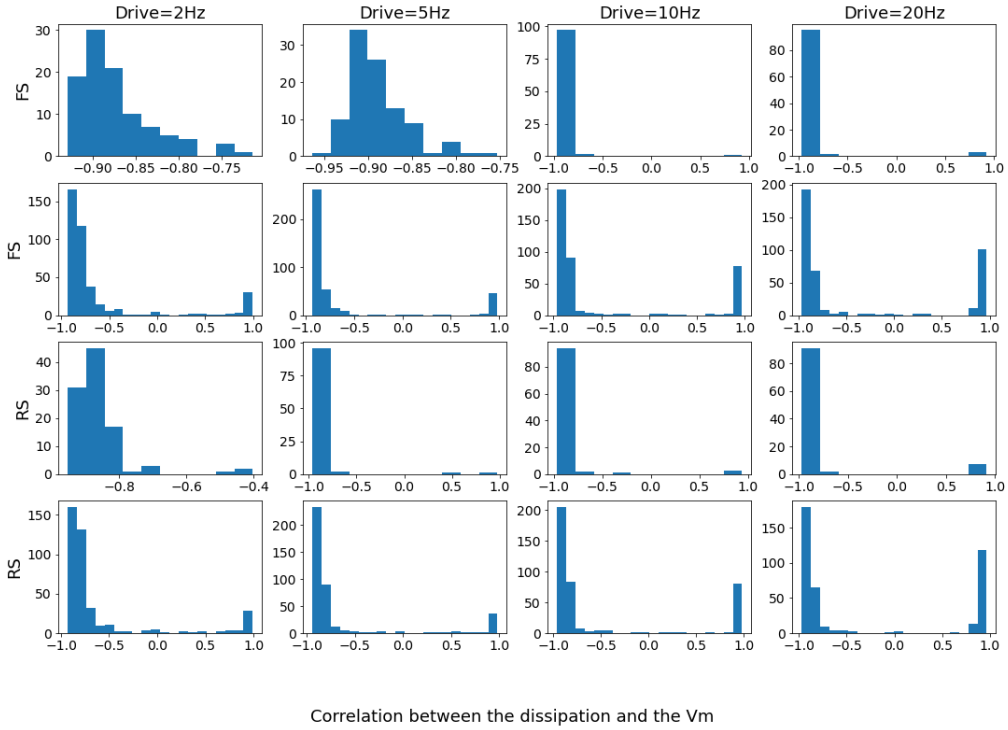


Figure 63: Distribution of the correlations between Vm and dissipation for each neuron from the previously defined lower subset for a whole simulation, for an HH network with various drives

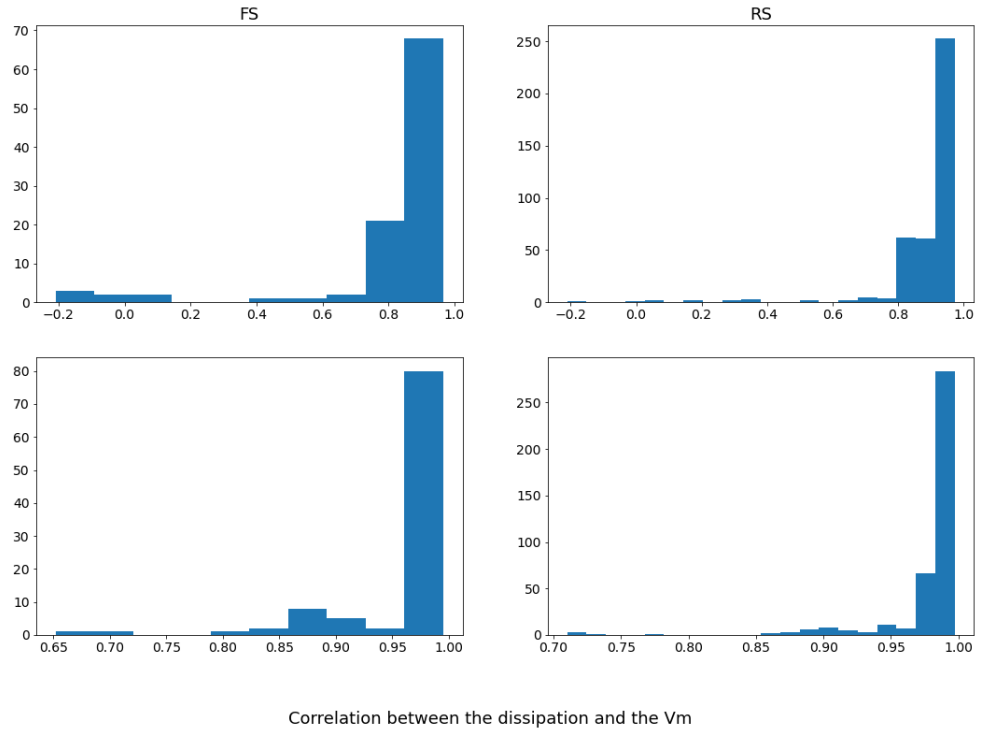


Figure 64: Distribution of the correlations between Vm and dissipation for each neuron from the previously defined higher subset for a whole simulation, for a self-sustained HH network

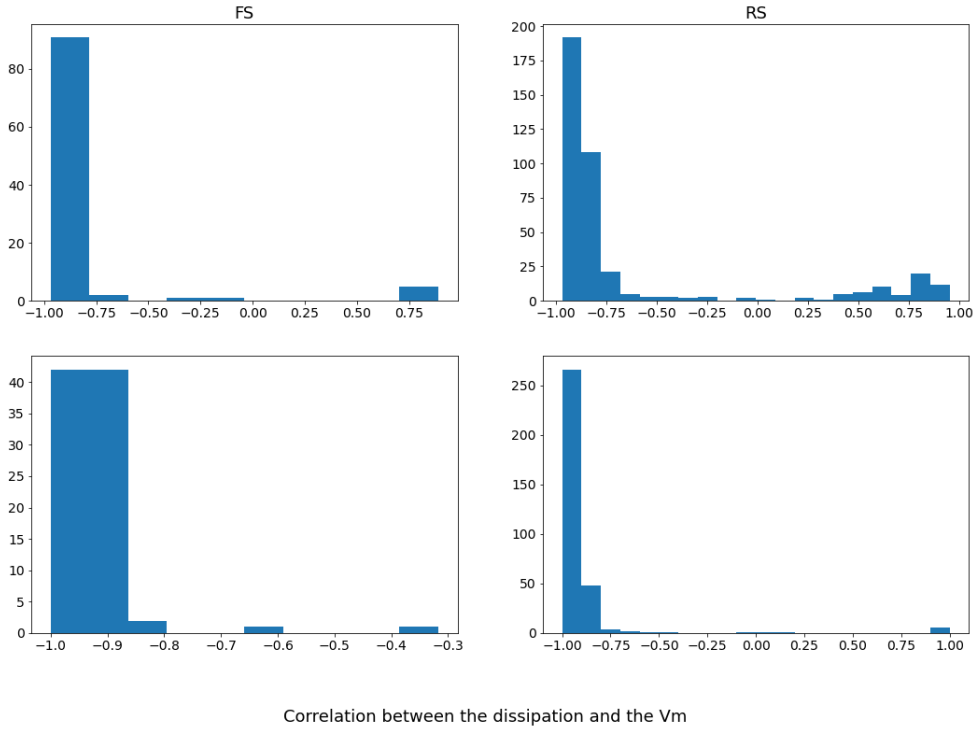


Figure 65: Distribution of the correlations between Vm and dissipation for each neuron from the previously defined lower subset for a whole simulation, for a self-sustained HH network

Our hypothesis was mostly what we found, as can be seen in Fig.62 and Fig.63.

The negative correlation is strong, and there is still some effect of the drive that allows to have some positive correlation (very high) for higher drive. RS is more polarized than FS. The positive effect is mostly the same but reversed, although the negative correlation when it happens is weaker and only appears for high drive.

The new self-sustained work even better, with very few "unwanted" correlations.

This allows us to say that there are indeed two regions of activity that are correlated differently for HH, and more work on the interpretation is required. It is worth to note that they don't perfectly split the correlations, so there are still some other factors.

**To conclude** , the investigation of the link between the dissipation and Vm seem like a promising one, as it allows us to go to the neuron scale, something that was not present enough in our general analysis. It also shows us more differences between AdEx and HH networks, helping us understand how their dynamics are functionally different as networks and not only as single neurons. Finally, it also gives us more information on the difference between Driven and Self-sustained system that where the core of chapter 3, showing us the investigation on those differences is far from over.

# 19 Codes

Finally, we will give here the two most useful codes in our analysis.

First, we give the code we use to simulate neural networks. Here, it is for a specific case of an AdEx Driven network, but a change of parameters can give different Driven or even Self-sustained networks, and a change in the differential equation and the associated parameters can give HH networks).

Then, we give the functions used to compute the FLE from the reconstructed system. Calling the function **LE_algo** with the appropriate parameters is all that is required to have the FLE as an output, similar to Fig.

## 19.1 Code for the simulation of an AdEx network

Here we give an example of a simulation of an AdEx driven network.

```
import matplotlib.pyplot as plt
import numpy as np
from brian2 import *
from time import time
import pickle



N1=2000
N2=8000
prbC=0.05*2000/N1



def bin_array(array, BIN, time_array):
N0 = int(BIN/(time_array[1]-time_array[0]))
N1 = int((time_array[-1]-time_array[0])/BIN)
return array[:N0*N1].reshape((N1,N0)).mean(axis=1)



def multi_Standard(input_freq,N1,N2,prbC,NbSims):
tic=time()
start_scope()
```

```
ResultsTotal=[]



for sim in NbSims:
print('Simulation #{} out of {}'.format(sim+1,NbSim))
DT=0.1
defaultclock.dt = DT*ms

TotTime=20000
duration = TotTime*ms

seed(sim)

eqs='''
dv/dt = (-GsynE*(v-Ee)-GsynI*(v-Ei)-gl*(v-El)+ gl*Dt*exp((v-Vt)/Dt)-w + Is)/Cm : volt (unless refrac
dw/dt = (a*(v-El)-w)/tau_w:ampere
dGsynI/dt = -GsynI/Tsyn : siemens
dGsynE/dt = -GsynE/Tsyn : siemens
Is:ampere
Cm:farad
gl:siemens
El:volt
a:siemens
tau_w:second
Dt:volt
Vt:volt
Ee:volt
Ei:volt
Tsyn:second
'''



# Population 1 - FS
b1 = 0.0*pA
G1 = NeuronGroup(N1, eqs, threshold='v > -47.5*mV', reset='v = -65*mV', refractory='5*ms', method='he
```

```
#init:
G1.v = -65*mV
G1.w = 0.0*pA
G1.GsynI=0.0*nS
G1.GsynE=0.0*nS
#parameters
G1.Cm = 200.*pF
G1.gl = 15.*nS
G1.El = -65.*mV
G1.Vt = -50.*mV
G1.Dt = 0.5*mV
G1.tau_w = 1.0*ms
G1.a = 0.0*nS
G1.Is = 0.0

G1.Ee=0.*mV
G1.Ei=-80.*mV
G1.Tsyn=5.*ms




# Population 2 - RS
b2 = 60.*pA
G2 = NeuronGroup(N2, eqs, threshold='v > -40.0*mV', reset='v = -55*mV; w += b2', refractory='5*ms',
G2.v = -65.*mV
G2.w = 0.0*pA
G2.GsynI=0.0*nS
G2.GsynE=0.0*nS
G2.Cm = 200.*pF
G2.gl = 15.*nS
G2.El = -65.*mV #before : -70mV
G2.Vt = -50.*mV
G2.Dt = 2.*mV
G2.tau_w = 500.*ms
```

```
G2.a = 0.*nS
G2.Is = 0.0*nA

G2.Ee=0.*mV
G2.Ei=-80.*mV
G2.Tsyn=5.*ms


W_tot = NeuronGroup(1,'W_tot : 1'  , method='heun')

# external drive-----------------------------------------------------------------------------
P_ed=PoissonGroup(8000, rates=input_freq*Hz)




# connections--------------------------------------------------------------------------------

Qi=5.0*nS
Qe=1.5*nS

prbC2=0.05

S_12 = Synapses(G1, G2, on_pre='GsynI_post+=Qi') #'v_post -= 1.*mV')
S_12.connect('i!=j', p=prbC2)

S_11 = Synapses(G1, G1, on_pre='GsynI_post+=Qi')
S_11.connect('i!=j',p=prbC2)

S_21 = Synapses(G2, G1, on_pre='GsynE_post+=Qe')
S_21.connect('i!=j',p=prbC)

S_22 = Synapses(G2, G2, on_pre='GsynE_post+=Qe')
S_22.connect('i!=j', p=prbC)
```

```
S_ed_in = Synapses(P_ed, G1, on_pre='GsynE_post+=Qe')
S_ed_in.connect(p=prbC2)

S_ed_ex = Synapses(P_ed, G2, on_pre='GsynE_post+=Qe')
S_ed_ex.connect(p=prbC2)#0.05)

#Connect the control neurons to the populations. To do so, apply the interesting function as the pos


S_W=Synapses(G2, W_tot, 'W_tot_post = w_pre : 1 (summed)')
S_W.connect(p=1)



################################################################################################

####        RECORDER GROUPS     ######

################################################################################################

dt_rec=1*ms


M1G1 = SpikeMonitor(G1)
FRG1 = PopulationRateMonitor(G1)

M1G2 = SpikeMonitor(G2)
FRG2 = PopulationRateMonitor(G2)

MonW_tot=StateMonitor(W_tot, 'W_tot', record=0)

print('--##Start simulation##--')
run(duration)
print('--##End simulation##--')
```

```
tac=time()
print('Simulation took {}s'.format(tac-tic))



LfrG1=np.array(FRG1.smooth_rate(window='gaussian', width=500*ms)/Hz)

LfrG2=np.array(FRG2.smooth_rate(window='gaussian', width=500*ms)/Hz)
WG2_tot=MonW_tot.W_tot[0]

#Save all

FR_input=[LfrG1,LfrG2]

ResultsTotal.append([FR_input,WG2_tot])



with open('Simulation_AdEx_1',"wb") as f:
pickle.dump(ResultsTotal,f)

input_freq=1.5
NbSim=[42]
multi_Standard(input_freq,N1,N2,prbC,NbSim)
```

## 19.2 Code for the lyapunov exponents in time series

Here we give the main functions used to compute the FLE from time series

```
import numpy as np
from copy import deepcopy as dc



def find_close_points(numX,data,delta,dim,Dt):
"""Find the label numY of a point Y such that dist(X,Y)<delta for X, Y in data.
```

```
If there is not at least 10 data points, return -1."""

X=np.array(data[numX])
Xmax=[]
for i in range(dim):
Xmax.append([X[i]-delta,X[i]+delta])
close_points=[]
flag=0
i=0
data_dt=data[:-Dt-1]

while i<len(data_dt):
val=data_dt[i]
i+=1
flag2=0
for j in range(dim):
if val[j]<Xmax[j][0] or val[j]>Xmax[j][1]:
flag2=1
break
if flag2==1:
continue
elif norm(X-val)<delta:
if i-1 not in range(numX-100,numX+100):
close_points.append(i-1)
flag+=1


if flag>1:
return close_points
else:
return -1


def find_points_ortho(numX,numY,data,delta,Dt,dim):
"""We have numX and numY, index of points X and Y that are respectively the
```

```
fiducial point and a point that evolved and is aligned with the FLE vector.
We want to find a point aligned with the vector (X,Y) but closer to X,
that we will call Y1. This will help to keep finding local values of the FLE. """
points=find_close_points(numX,data,delta,dim,Dt)
angle1=0.95#min cos(angle) we allow
flag1=0
new_num1=-1
X=np.array(data[numX])
Y=np.array(data[numY])
v1=Y-X
delta_temp=delta
while points==-1:
points=find_close_points(numX,data,delta_temp,dim,Dt)
delta_temp+=0.5*delta
delta_temp1=delta_temp
points1=dc(points)
#Let's select the vector that aligns with the FLE
while flag1==0:
for num in points1:
v2=data[num]-X
ang=angle_between(v1,v2)#we want something as close to 1 as possible (colinear)
if ang>angle1:
flag1=1
angle1=ang
new_num1=num
if flag1==0:
delta_temp1+=0.5*delta
points1_2=find_close_points(numX,data,delta_temp1,dim,Dt)
points1=[val for val in points1_2 if val not in points1]

return new_num1



def LE_algo(Xs,t0,Dt,dt,delta,eps0,time_max):
```

```
"""Standard algo for lyapunov spectrum with Dt a time between each
LE evaluation, dt a time step, eps a small perturbation, eps0 a tolerance
to end the algo, t0 the initial value x(t0) and u0 a set of orthonormal
vectors at time T0. Here we use time series instead of the flow. The main difference
is therefore that we use closer neighbors instead of perturbations."""

dim=np.shape(Xs)[1]

l_pert1=[0] #initialize the LE perturbated

#initialize the first point that will align with the FLE
ty0=-1
delta_temp=delta
while ty0==-1:
ty0=find_close_points(t0,Xs[:-Dt-1],delta_temp,dim,Dt)
delta_temp+=0.5*delta
ty0=np.random.permutation(ty0[:-1])[-1]

for t in range(time_max): #I prefer long time to while loops
#Evolution of the trajectory with a step dt until Dt

x0=np.array(Xs[t0])



#"Perturbation" like (finding points close and colinear to the vectors (x1,y1),
#and another one orthogonal to it)
x_pert1_nums=find_points_ortho(t0,ty0,Xs,delta,Dt,dim)
dists=norm(Xs[x_pert1_nums]-x0)
#2D distances between original points and "perturbation"

#"Evolve perturbation"
Dt_temp=0
for i in range(Dt):
Dt_temp+=1
if (x_pert1_nums+Dt_temp)>=len(Xs)-Dt:
```

```
Dt_temp-=1
break



t1=t0+Dt_temp
x1=Xs[t1]

x_pert1_nums=x_pert1_nums+Dt_temp#2D index
x_pert1=Xs[x_pert1_nums] #2 3D-coordinates

#Compute difference
w1=x_pert1-x1



#Find growth
v_norms1=norm(w1)#gives the norm of the first vector and the norm of the projection
#Compute the LS
l_pert1.append(np.log(v_norms1/dists)/(Dt_temp*dt))
#Check if we can stop
if np.abs(np.mean(l_pert1[:-1])-np.mean(l_pert1[:-2]))<eps0:
return t+1,l_pert1
else:
t0=t1
ty0=x_pert1_nums
return -1,l_pert1
```

# References


Alreja, A., Nemenman, I., & Rozell, C. (2022, 01). Constrained brain volume in an efficient coding model explains the fraction of excitatory and inhibitory neurons in sensory cortices. *PLOS Computational Biology*, *18*, e1009642. doi: 10.1371/journal.pcbi.1009642

Amit, D., & Brunel, N. (1997, 04). Model of global spontaneous activity and local structured activity during delay periods in the cerebral cortex cerebral cortex 7 237-52. *Cerebral cortex (New York, N.Y. : 1991)*, *7*, 237-52. doi: 10.1093/cercor/7.3.237

Anderson, W., Kudela, P., Cho, J., Bergey, G., & Franaszczuk, P. (2007, 09). Studies of stimulus

parameters for seizure disruption using neural network simulations. *Biological cybernetics*, *97*, 173-94. doi: 10.1007/s00422-007-0166-0

Azarfar, A., Calcini, N., Huang, C., Zeldenrust, F., & Celikel, T. (2018). Neural coding: A single neuron's perspective. *Neuroscience & Biobehavioral Reviews*, *94*, 238-247. Retrieved from https://www.sciencedirect.com/science/article/pii/S0149763417308941 doi: https://doi.org/10.1016/j.neubiorev.2018.09.007

Beghi, E. (2019, December). The epidemiology of epilepsy. *Neuroepidemiology*, *54*(Suppl. 2), 185–191. Retrieved from https://doi.org/10.1159/000503831 doi: 10.1159/000503831

Boon, P., Cock, E. D., Mertens, A., & Trinka, E. (2018, April). Neurostimulation for drug-resistant epilepsy. *Current Opinion in Neurology*, *31*(2), 198–210. Retrieved from https://doi.org/10.1097/wco.0000000000000534 doi: 10.1097/wco.0000000000000534

Borst, A., & Theunissen, F. E. (1999). Information theory and neural coding. *Nature Neuroscience*, *2*, 947-957.

Bouté, J., & Destexhe, A. (2023). Dynamical properties of self-sustained and driven neural networks. *bioRxiv*. Retrieved from https://www.biorxiv.org/content/early/2023/07/18/2023.07.15.549166 doi: 10.1101/2023.07.15.549166

Brette, R., & Gerstner, W. (2005a, November). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology*, *94*(5), 3637–3642. doi: 10.1152/jn.00686.2005

Brette, R., & Gerstner, W. (2005b, November). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *Journal of Neurophysiology*, *94*(5), 3637–3642. Retrieved from https://doi.org/10.1152/jn.00686.2005 doi: 10.1152/jn.00686.2005

Broomhead, D. S., & Jones, R. L. (1989). Time-series analysis. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, *423*, 103 - 121.

Brunel, N. (2000a). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of computational neuroscience*, *8*(3), 183–208.

Brunel, N. (2000b, 01). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of Computational Neuroscience*, *8*. doi: 10.1023/A:1008925309027

Capone, C., Volo, M., Romagnoni, A., Mattia, M., & Destexhe, A. (2019, 12). State-dependent mean-field formalism to model different activity states in conductance-based networks of spiking neurons. *Physical Review E*, *100*. doi: 10.1103/PhysRevE.100.062413

Carlu, M., Chehab, O., Dalla Porta, L., Depannemaecker, D., Héricé, C., Jedynak, M., ... Volo, M. (2019, 12). A mean-field approach to the dynamics of networks of complex neurons, from nonlinear integrate-and-fire to hodgkin-huxley models. *Journal of Neurophysiology*, *123*. doi:

10.1152/jn.00399.2019

Carlu, M., Chehab, O., Porta, L. D., Depannemaecker, D., Héricé, C., Jedynak, M., ... di Volo, M. (2020, March). A mean-field approach to the dynamics of networks of complex neurons, from nonlinear integrate-and-fire to hodgkin–huxley models. *Journal of Neurophysiology*, *123*(3), 1042–1051. Retrieved from `https://doi.org/10.1152/jn.00399.2019` doi: 10.1152/jn.00399.2019

Cessac, B. (2008, 04). A discrete time neural network model with spiking neurons : Rigorous results on the spontaneous dynamics. *Journal of mathematical biology*, *56*, 311-45. doi: 10.1007/s00285-007-0117-3

Cessac, B. (2009, 01). A view of neural networks as dynamical systems. *International Journal of Bifurcation and Chaos*, *20*. doi: 10.1142/S0218127410026721

Cessac, B. (2019, 10). Linear response in neuronal networks: From neurons dynamics to collective response. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *29*(10), 103105. Retrieved from `https://doi.org/10.1063/1.5111803` doi: 10.1063/1.5111803

Cessac, B., & Viéville, T. (2008, 02). On dynamics of integrate-and-fire neural networks with conductance based synapses. *Frontiers in computational neuroscience*, *2*, 2. doi: 10.3389/neuro.10.002.2008

Chalmers, A. (2013). *What is this thing called science? 4th edition*. Hackett Publishing Company.

Chow, M., Wu, S., Webb, S., Gluskin, K., & Yew, D. (2017, 01). Functional magnetic resonance imaging and the brain: A brief review. *World Journal of Radiology*, *9*, 5. doi: 10.4329/wjr.v9.i1.5

Cofre, R., & Cessac, B. (2014, 05). Exact computation of the maximum entropy potential of spiking neural networks models. *Phys. Rev. E*, *89*. doi: 10.1103/PhysRevE.89.052117

Coombes, S. (1999, 05). Liapunov exponents and mode-locked solutions for integrate-and-fire dynamical systems. *Physics Letters A*, *255*, 49-57. doi: 10.1016/S0375-9601(99)00172-3

Davis, Z., Benigno, G., Fletterman, C., Desbordes, T., Sejnowski, T., Reynolds, J., & Muller, L. (2020, 11). Spontaneous traveling waves naturally emerge from horizontal fiber time delays and travel through locally asynchronous-irregular states. *Nature Communications*. doi: 10.21203/rs.3.rs-113978/v1

Deco, G., Lynn, C., Perl, Y. S., & Kringelbach, M. L. (2023). *Violations of the fluctuation-dissipation theorem reveal distinct non-equilibrium dynamics of brain states.*

Dehghani, N., Peyrache, A., Telenczuk, B., Le Van Quyen, M., Halgren, E., Cash, S., ... Destexhe, A. (2016, 03). Dynamic balance of excitation and inhibition in human and monkey neocortex. *Scientific Reports*, *6*, 23176. doi: 10.1038/srep23176

Dehghani, N., Peyrache, A., Telenczuk, B., Quyen, M. L. V., Halgren, E., Cash, S. S., ... Destexhe, A. (2016). Dynamic balance of excitation and inhibition in human and monkey neocortex. *Scientific Reports*, *6*(1). doi: 10.1038/srep23176

Depannemaecker, D., Carlu, M., Bouté, J., & Destexhe, A. (2022, 11). A model for the prop-

agation of seizure activity in normal brain tissue. *eneuro*, *9*, ENEURO.0234-21.2022. doi: 10.1523/ENEURO.0234-21.2022

Depannemaecker, D., Destexhe, A., Jirsa, V., & Bernard, C. (2021, February). Modeling seizures: From single neurons to networks. *preprint*. Retrieved from https://doi.org/10.20944/preprints202102.0478.v1 doi: 10.20944/preprints202102.0478.v1

Depannemaecker, D., Ivanov, A., Lillo, D., Spek, L., Bernard, C., & Jirsa, V. (2020). A unified physiological framework of transitions between seizures, sustained ictal activity and depolarization block at the single neuron level. *J Comput Neurosci*. Retrieved from https://doi.org/10.1101/2020.10.23.352021 doi: 10.1101/2020.10.23.352021

Destexhe, A. (2009a, June). Self-sustained asynchronous irregular states and up and down states in thalamic, cortical and thalamocortical networks of nonlinear integrate-and-fire neurons. *Journal of Computational Neuroscience*, *27*(3), 493–506. Retrieved from https://doi.org/10.1007/s10827-009-0164-4 doi: 10.1007/s10827-009-0164-4

Destexhe, A. (2009b, 07). Self-sustained asynchronous irregular states and up–down states in thalamic, cortical and thalamocortical networks of nonlinear integrate-and-fire neurons. *Journal of computational neuroscience*, *27*, 493-506. doi: 10.1007/s10827-009-0164-4

Destexhe, A., Rudolph, M., & Paré, D. (2003, 09). The high-conductance state of neocortical neurons in vivo. *Nature Reviews Neuroscience*, *4*. doi: 10.1038/nrn1289

di Volo, M., & Destexhe, A. (2021, 09). Optimal responsiveness and information flow in networks of heterogeneous neurons. *Scientific Reports*, *11*. doi: 10.1038/s41598-021-96745-2

Eckmann, J.-P., Kamphorst, S., Ruelle, D., & Ciliberto, S. (1987, 01). Liapunov exponents from time series. *Physical review. A*, *34*, 4971-4979. doi: 10.1103/PhysRevA.34.4971

Eckmann, J. P., & Ruelle, D. (1985, July). Ergodic theory of chaos and strange attractors. *Reviews of Modern Physics*, *57*(3), 617–656. Retrieved 2022-08-20, from https://link.aps.org/doi/10.1103/RevModPhys.57.617 (Publisher: American Physical Society) doi: 10.1103/RevModPhys.57.617

El-Boustani, S., & Destexhe, A. (2009, 02). A master equation formalism for macroscopic modeling of asynchronous irregular activity states. *Neural computation*, *21*, 46-100. doi: 10.1162/neco.2009.02-08-710

El Boustani, S., Pospischil, M., Rudolph-Lilith, M., & Destexhe, A. (2007). Activated cortical states: experiments, analyses and models. *Journal of Physiology-Paris*, *101*(1-3), 99–109.

Faugeras, O., & MacLaurin, J. (2015). Asymptotic description of neural networks with correlated synaptic weights. *Entropy*, *17*(7), 4701–4743. Retrieved from https://www.mdpi.com/1099-4300/17/7/4701 doi: 10.3390/e17074701

Faure, P., & Korn, H. (2001, September). Is there chaos in the brain? I. Concepts

of nonlinear dynamics and methods of investigation. *Comptes Rendus de l'Académie des Sciences - Series III - Sciences de la Vie*, *324*(9), 773–793. Retrieved 2021-10-19, from https://www.sciencedirect.com/science/article/pii/S0764446901013774 doi: 10.1016/S0764-4469(01)01377-4

Frigg, R., & Hartmann, S. (2020). Models in Science. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Spring 2020 ed.). Metaphysics Research Lab, Stanford University. Retrieved 2022-08-20, from https://plato.stanford.edu/archives/spr2020/entries/models-science/

Fu, T.-C. (2011, 02). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, *24*, 164-181. doi: 10.1016/j.engappai.2010.09.007

Gallavotti, G. (2004, 10). Entropy production and thermodynamics of nonequilibrium stationary states: A point of view. *Chaos (Woodbury, N.Y.)*, *14*, 680-90. doi: 10.1063/1.1781911

Gaspard, P. (2020, 11). Stochastic approach to entropy production in chemical chaos. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *30*(11), 113103. Retrieved from https://doi.org/10.1063/5.0025350 doi: 10.1063/5.0025350

Goldman, J., Kusch, L., Aquilue, D., Yalçınkaya, B. H., Depannemaecker, D., Ancourt, K., ... Destexhe, A. (2023, 01). A comprehensive neural simulation of slow-wave sleep and highly responsive wakefulness dynamics. *Frontiers in Computational Neuroscience*, *16*, 1058957. doi: 10.3389/fncom.2022.1058957

Górski, T., Depannemaecker, D., & Destexhe, A. (2021). Conductance-based adaptive exponential integrate-and-fire model. *Neural Computation*, *33*(1), 41–66. Retrieved from https://doi.org/10.1162/neco_a_01342 doi: 10.1162/neco_a_01342

Gross, J. (2019, 10). Magnetoencephalography in cognitive neuroscience: A primer. *Neuron*, *104*, 189-204. doi: 10.1016/j.neuron.2019.07.001

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory.* New York: Wiley. Hardcover.

Herzog, R., Mediano, P., Rosas, F., Lodder, P., Carhart-Harris, R., Perl, Y., ... Cofre, R. (2023, 04). A whole-brain model of the neural entropy increase elicited by psychedelic drugs. *Scientific Reports*. doi: 10.1038/s41598-023-32649-7

Hille, B. (1992). *Ionic channels of excitable membranes.* Sunderland, Mass: Sinauer Associates.

Hodgkin, A. L., & Huxley, A. F. (1952a, August). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, *117*(4), 500–544. Retrieved 2022-08-20, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1392413/

Hodgkin, A. L., & Huxley, A. F. (1952b, August). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, *117*(4), 500–544. Retrieved from https://doi.org/10.1113/jphysiol.1952.sp004764 doi: 10.1113/jphys-

iol.1952.sp004764

Härdle, W., Lütkepohl, H., & Chen, R. (1997). A review of nonparametric time series analysis. *International Statistical Review*, *65*(1), 49-72. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1751-5823.1997.tb00367.x doi: https://doi.org/10.1111/j.1751-5823.1997.tb00367.x

Izhikevich, E. (2004, September). Which Model to Use for Cortical Spiking Neurons? *IEEE Transactions on Neural Networks*, *15*(5), 1063–1070. Retrieved 2022-08-20, from http://ieeexplore.ieee.org/document/1333071/ doi: 10.1109/TNN.2004.832719

Izhikevich, E. M. (2007). *Dynamical Systems in Neuroscience*. MIT Press.

Jirsa, V. K., Stacey, W. C., Quilichini, P. P., Ivanov, A. I., & Bernard, C. (2014, 8). On the nature of seizure dynamics. *Brain : a journal of neurology*, *137*(Pt 8), 2210–30. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/24919973 http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4107736 doi: 10.1093/brain/awu133

Jiruska, P., de Curtis, M., Jefferys, J. G. R., Schevon, C. A., Schiff, S. J., & Schindler, K. (2013, January). Synchronization and desynchronization in epilepsy: controversies and hypotheses. *The Journal of Physiology*, *591*(4), 787–797. Retrieved from https://doi.org/10.1113/jphysiol.2012.239590 doi: 10.1113/jphysiol.2012.239590

Johnston D, W. S.-S. (1994). *Foundations of cellular neurophysiology*. The MIT Press.

Korn, H., & Faure, P. (2003). Is there chaos in the brain? II. Experimental evidence and related models. *Comptes rendus biologies*. doi: 10.1016/J.CRVI.2003.09.011

Kuznetsov, N., Mokaev, T., Kuznetsova, O., & Kudryashova, E. (2020, 10). The lorenz system: hidden boundary of practical stability and the lyapunov dimension. *Nonlinear Dynamics*, *102*, 1-20. doi: 10.1007/s11071-020-05856-4

Köppen, M. (2000). The curse of dimensionality. In *5th online world conference on soft computing in industrial applications (WSC5)* (Vol. 1, pp. 4–8).

Lagarde, S., Buzori, S., Trebuchon, A., Carron, R., Scavarda, D., Milh, M., ... Bartolomei, F. (2018, November). The repertoire of seizure onset patterns in human focal epilepsies: Determinants and prognostic values. *Epilepsia*, *60*(1), 85–95. Retrieved from https://doi.org/10.1111/epi.14604 doi: 10.1111/epi.14604

Layer, M., Senk, J., Essink, S., van Meegen, A., Bos, H., & Helias, M. (2022). Nnmt: Mean-field based analysis tools for neuronal network models. *Frontiers in Neuroinformatics*, *16*. Retrieved from https://www.frontiersin.org/articles/10.3389/fninf.2022.835657 doi: 10.3389/fninf.2022.835657

Lee, A. K., Manns, I. D., Sakmann, B., & Brecht, M. (2006). Whole-cell

recordings in freely moving rats. *Neuron*, *51*(4), 399-407. Retrieved from https://www.sciencedirect.com/science/article/pii/S0896627306005435 doi: https://doi.org/10.1016/j.neuron.2006.07.004

Leonov, G., Kuznetsov, N., Korzhemanova, N., & Kusakin, D. (2015, 08). The lyapunov dimension formula for the global attractor of the lorenz system. *Communications in Nonlinear Science and Numerical Simulation*, *41*. doi: 10.1016/j.cnsns.2016.04.032

Lindner, B. (2022, Oct). Fluctuation-dissipation relations for spiking neurons. *Phys. Rev. Lett.*, *129*, 198101. Retrieved from https://link.aps.org/doi/10.1103/PhysRevLett.129.198101 doi: 10.1103/PhysRevLett.129.198101

Lorenz, E. N. (2004). Deterministic nonperiodic flow. In B. R. Hunt, T.-Y. Li, J. A. Kennedy, & H. E. Nusse (Eds.), *The theory of chaotic attractors* (pp. 25–36). New York, NY: Springer New York. Retrieved from https://doi.org/10.1007/978-0-387-21830-4_2 doi: 10.1007/978-0-387-21830-4_2

Markram, H., Muller, E., Ramaswamy, S., Reimann, M. W., Abdellah, M., Sanchez, C. A., ... Schürmann, F. (2015, October). Reconstruction and Simulation of Neocortical Microcircuitry. *Cell*, *163*(2), 456–492. Retrieved 2022-08-20, from https://www.cell.com/cell/abstract/S0092-8674(15)01191-5 (Publisher: Elsevier) doi: 10.1016/j.cell.2015.09.029

Martin, J. H. (2012). *Neuroanatomy text and atlas, 4e.* McGraw Hill / Medical.

Matsumura, M., Cope, T., & Fetz, E. (1988, 02). Sustained excitatory synaptic input to motor cortex neurons in awake animals revealed by intracellular recording of membrane potentials. *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale*, *70*, 463-9. doi: 10.1007/BF00247594

Michel, C., & Brunet, D. (2019, 04). Eeg source imaging: A practical review of the analysis steps. *Frontiers in Neurology*, *10*. doi: 10.3389/fneur.2019.00325

Mountcastle, V. B. (1957). Modality and topographic properties of single neurons of cat's somatic sensory cortex. *Journal of neurophysiology*, *20 4*, 408-34.

Muller, L., Reynaud, A., Chavane, F., & Destexhe, A. (2014, 04). The stimulus-evoked population response in visual cortex of awake monkey is a propagating wave. *Nature communications*, *5*, 3675. doi: 10.1038/ncomms4675

Nadler, J. V., & Spencer, D. D. (2014). What is a seizure focus? *Adv Exp Med Biol*, 55–62. Retrieved from https://doi.org/10.1007/978-94-017-8914-1_4 doi: 10.1007/978-94-017-8914-1_4

Naud, R., Marcille, N., Clopath, C., & Gerstner, W. (2008a). Firing patterns in the adaptive exponential integrate-and-fire model. *Biological Cybernetics*, *99*(4), 335–347. Retrieved 2022-08-20, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2798047/ doi: 10.1007/s00422-008-0264-7

Naud, R., Marcille, N., Clopath, C., & Gerstner, W. (2008b). Firing patterns in the adaptive exponential integrate-and-fire model. *Biological cybernetics*, *99*(4-5), 335.

Neumann, A. R., Raedt, R., Steenland, H. W., Sprengers, M., Bzymek, K., Navratilova, Z., ... Luczak, A. (2017, August). Involvement of fast-spiking cells in ictal sequences during spontaneous seizures in rats with chronic temporal lobe epilepsy. *Brain*, *140*(9), 2355–2369. Retrieved from `https://doi.org/10.1093/brain/awx179` doi: 10.1093/brain/awx179

Oseledets, V. (1968). A multiplicative ergodic theorem.liapunov characteristic number for dynamical systems. *Trans.Moscow Math.Soc.*, *19*, 197-231.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., & Shaw, R. S. (1980, Sep). Geometry from a time series. *Phys. Rev. Lett.*, *45*, 712–716. Retrieved from `https://link.aps.org/doi/10.1103/PhysRevLett.45.712` doi: 10.1103/PhysRevLett.45.712

Pais-Vieira, M., Yadav, A. P., Moreira, D., Guggenmos, D., Santos, A., Lebedev, M., & Nicolelis, M. A. L. (2016, September). A closed loop brain-machine interface for epilepsy control using dorsal column electrical stimulation. *Scientific Reports*, *6*(1). Retrieved from `https://doi.org/10.1038/srep32814` doi: 10.1038/srep32814

Paulk, A. C., Kfir, Y., Khanna, A. R., Mustroph, M. L., Trautmann, E. M., Soper, D. J., ... Cash, S. S. (2022, jan). Large-scale neural recordings with single neuron resolution using neuropixels probes in human cortex. *Nature Neuroscience*, *25*(2), 252–263. Retrieved from `https://doi.org/10.1038%2Fs41593-021-00997-0` doi: 10.1038/s41593-021-00997-0

Pchelintsev, A. (2014, 04). Numerical and physical modeling of the dynamics of the lorenz system. *Numerical Analysis and Applications*, *7*, 159-167. doi: 10.1134/S1995423914020098

Peyrache, A., Dehghani, N., Eskandar, E. N., Madsen, J. R., Anderson, W. S., Donoghue, J. A., ... Destexhe, A. (2012, January). Spatiotemporal dynamics of neocortical excitation and inhibition during human sleep. *Proceedings of the National Academy of Sciences*, *109*(5), 1731–1736. Retrieved from `https://doi.org/10.1073/pnas.1109895109` doi: 10.1073/pnas.1109895109

Pikovsky, A., & Politi, A. (2016). *Lyapunov Exponents: A Tool to Explore Complex Dynamics*. Cambridge University Press. (Google-Books-ID: krOnCwAAQBAJ)

Rajna, P., & Lona, C. (1989, April). Sensory stimulation for inhibition of epileptic seizures. *Epilepsia*, *30*(2), 168–174. Retrieved from `https://doi.org/10.1111/j.1528-1157.1989.tb05450.x` doi: 10.1111/j.1528-1157.1989.tb05450.x

Ruelle, D. (1989). *Chaotic evolution and strange attractors*. Cambridge University Press. doi: 10.1017/CBO9780511608773

Ruelle, D. (1996, October). Positivity of entropy production in nonequilibrium statistical mechanics. *Journal of Statistical Physics*, *85*(1-2), 1-23. doi: 10.1007/BF02175553

Ruelle, D. (2003, 04). Extending the definition of entropy to nonequilibrium steady states. *Proceed-*

*ings of the National Academy of Sciences of the United States of America*, *100*, 3054-8. doi: 10.1073/pnas.0630567100

Ruelle, D. (2009, mar). A review of linear response theory for general differentiable dynamical systems. *Nonlinearity*, *22*(4), 855. Retrieved from https://dx.doi.org/10.1088/0951-7715/22/4/009 doi: 10.1088/0951-7715/22/4/009

Saggio, M. L., Crisp, D., Scott, J. M., Karoly, P., Kuhlmann, L., Nakatani, M., ... Stacey, W. C. (2020, July). A taxonomy of seizure dynamotypes. *eLife*, *9*. Retrieved from https://doi.org/10.7554/elife.55632 doi: 10.7554/elife.55632

Saggio, M. L., Spiegler, A., Bernard, C., & Jirsa, V. K. (2017). Fast–Slow Bursters in the Unfolding of a High Codimension Singularity and the Ultra-slow Transitions of Classes. *The Journal of Mathematical Neuroscience*, *7*(1), 7. Retrieved from https://doi.org/10.1186/s13408-017-0050-8 doi: 10.1186/s13408-017-0050-8

Sanz-Leon, P., Knock, S., Spiegler, A., & Jirsa, V. (2015, 01). Mathematical framework for large-scale brain network modeling in the virtual brain. *NeuroImage*, *111*. doi: 10.1016/j.neuroimage.2015.01.002

Soltesz, I., & Staley, K. (Eds.). (2008). *Computational neuroscience in epilepsy*. Elsevier. Retrieved from https://doi.org/10.1016/b978-0-12-373649-9.x5001-7 doi: 10.1016/b978-0-12-373649-9.x5001-7

Sompolinsky, H., Crisanti, A., & Sommers, H. J. (1988, Jul). Chaos in random neural networks. *Phys. Rev. Lett.*, *61*, 259–262. Retrieved from https://link.aps.org/doi/10.1103/PhysRevLett.61.259 doi: 10.1103/PhysRevLett.61.259

Sparrow, C. (1982). The lorenz equations: Bifurcations, chaos, and strange attractors. *Applied Mathematical Sciences*, *41*. doi: https://doi.org/10.1007/978-1-4612-5767-7

Steriade, M., Timofeev, I., & Grenier, F. (2001, 05). Natural waking and sleep states: A view from inside neocortical neurons. *J Neurophysiol*, *85*, 1969-85. doi: 10.1152/jn.2001.85.5.1969

Sterratt, D., Graham, B., Gillies, A., & Willshaw, D. (2011). *Principles of computational modelling in neuroscience*. doi: 10.1017/CBO9780511975899

Strogatz, S. H. (2019). *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering* (2nd ed.). Boca Raton: CRC Press. doi: 10.1201/9780429492563

Takens, F. (1981). Detecting strange attractors in turbulence. *Dynamical Systems and Turbulence, Lecture Notes in Mathematics*, *898*, 366–381. doi: https://doi.org/10.1007/BFb0091924

Taylor, P. N., Wang, Y., Goodfellow, M., Dauwels, J., Moeller, F., Stephani, U., & Baier, G. (2014, December). A computational study of stimulus driven epileptic seizure abatement. *PLoS ONE*, *9*(12), e114316. Retrieved from https://doi.org/10.1371/journal.pone.0114316 doi: 10.1371/journal.pone.0114316

Torao-Angosto, M., Manasanch, A., Mattia, M., & Sanchez-Vives, M. (2021, 02). Up and down states during slow oscillations in slow-wave sleep and different levels of anesthesia. *Frontiers in Systems Neuroscience*, *15*. doi: 10.3389/fnsys.2021.609645

Tél, T., & Lai, Y.-C. (2008). Chaotic transients in spatially extended systems. *Physics Reports*, *460*(6), 245-275. Retrieved from https://www.sciencedirect.com/science/article/pii/S0370157308000379 doi: https://doi.org/10.1016/j.physrep.2008.01.001

van Meegen, A., Kühn, T., & Helias, M. (2021, Oct). Large-deviation approach to random recurrent neuronal networks: Parameter inference and fluctuation-induced transitions. *Phys. Rev. Lett.*, *127*, 158302. Retrieved from https://link.aps.org/doi/10.1103/PhysRevLett.127.158302 doi: 10.1103/PhysRevLett.127.158302

van Vreeswijk, C., & Sompolinsky, H. (1996). Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science*, *274*(5293), 1724-1726. Retrieved from https://www.science.org/doi/abs/10.1126/science.274.5293.1724 doi: 10.1126/science.274.5293.1724

Viana, M. (2000). What's new on lorenz strange attractors? *The Mathematical Intelligencer*, *22*, 6-19.

Vogels, T., & Abbott, L. (2005, 12). Signal propagation and logic gating in networks of integrate-and-fire neurons. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, *25*, 10786-95. doi: 10.1523/JNEUROSCI.3508-05.2005

Volo, M., Romagnoni, A., Capone, C., & Destexhe, A. (2019, 02). Biologically realistic mean-field models of conductance-based networks of spiking neurons with adaptation. *Neural Computation*, *31*, 1-28.

von Bartheld, C. S., Bahney, J., & Herculano-Houzel, S. (2016). The search for true numbers of neurons and glial cells in the human brain: A review of 150 years of cell counting. *Journal of Comparative Neurology*, *524*.

Vreeswijk, C. v., & Sompolinsky, H. (1998). Chaotic balanced state in a model of cortical circuits. *Neural Computation*, *10*(6), 1321-1371. doi: 10.1162/089976698300017214

Wilson, H., & Cowan, J. (1972, January). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal*, *12*(1), 1-24. doi: 10.1016/S0006-3495(72)86068-5

Wolf, A., Swift, J., Swinney, H., & Vastano, J. (1985, 07). Determining lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, *16*, 285-317. doi: 10.1016/0167-2789(85)90011-9

Wonders, C., & Anderson, S. (2006, 10). The origin and specification of cortical interneurons. *Nature reviews. Neuroscience*, *7*, 687-96. doi: 10.1038/nrn1954

Ye, H., & Kaszuba, S. (2019, December). Neuromodulation with electromagnetic stimulation for seizure suppression: From electrode to magnetic coil. *IBRO Reports*, *7*, 26–33. Retrieved from https://doi.org/10.1016/j.ibror.2019.06.001 doi: 10.1016/j.ibror.2019.06.001

Yger, P., El-Boustani, S., Destexhe, A., & Frégnac, Y. (2011, 10). Topologically invariant macroscopic statistics in balanced networks of conductance-based integrate-and-fire neurons. *Journal of computational neuroscience*, *31*, 229-45. doi: 10.1007/s10827-010-0310-z

Yu, W., Chen, G., Cao, J., Lü, J., & Parlitz, U. (2007, Jun). Parameter identification of dynamical systems from time series. *Phys. Rev. E*, *75*, 067201. Retrieved from `https://link.aps.org/doi/10.1103/PhysRevE.75.067201` doi: 10.1103/PhysRevE.75.067201

Zerlaut, Y., Chemla, S., Chavane, F., & Destexhe, A. (2017, 11). Modeling mesoscopic cortical dynamics using a mean-field model of conductance-based networks of adaptive exponential integrate-and-fire neurons. *J. Comput. Neurosci.*. doi: 10.1101/168385

Zerlaut, Y., Chemla, S., Chavane, F., & Destexhe, A. (2018, November). Modeling mesoscopic cortical dynamics using a mean-field model of conductance-based networks of adaptive exponential integrate-and-fire neurons. *Journal of Computational Neuroscience*, *44*(1), 45–61. Retrieved from `https://doi.org/10.1007/s10827-017-0668-2` doi: 10.1007/s10827-017-0668-2

Zerlaut, Y., & Destexhe, A. (2017, 06). Enhanced responsiveness and low-level awareness in stochastic network states. *Neuron*, *94*, 1002-1009. doi: 10.1016/j.neuron.2017.04.001