



UNIVERSITY OF CORDOBA
INSTITUTE OF POSTGRADUATE STUDIES



UNIVERSITY MASTER'S DEGREE

**PHYSICAL TECHNOLOGY:
RESEARCH AND APPLICATIONS**

MASTER'S THESIS

**Exploring the relationship between Hawkes processes
and self-organized criticality in living systems**

Antonio Rivas Blanco

Tutor(s): Jorge Hidalgo Aguilera and Serena di Santo

Line of research: Modelling of complex systems and their interdisciplinary applications

Córdoba, 07/2024

Autorización de defensa del Trabajo Fin de Máster

Tutor 1: Jorge Hidalgo Aguilera

Tutor 2: Serena di Santo

INFORMAN: Que el presente trabajo titulado *Exploring the relationship between Hawkes processes and self-organized criticality in living systems* que constituye la memoria del Trabajo Fin de Máster ha sido realizado por Antonio Rivas Blanco y AUTORIZA/N su presentación para que pueda ser defendido en la convocatoria 1ª con fecha 23/07/2024.

Firmado en Córdoba a 11 de julio de 2024

Tutor 1: Jorge Hidalgo Aguilera

Tutor 2: Serena di Santo

Declaración de autoría

Nombre y apellidos: Antonio Rivas Blanco
DNI: 49832223D

DECLARA

1. Que el trabajo que presenta es totalmente original y que se hace responsable de todo su contenido.
2. Que no ha sido publicado ni total ni parcialmente.
3. Que todos los aportes de otros autores/as han sido debidamente referenciados.
4. Que no ha incurrido en fraude científico, plagio o vicios de autoría.
5. Que, en caso de no cumplir los requerimientos anteriores, aceptará las medidas disciplinarias sancionadoras que correspondan.

Firmado en Córdoba a, 11 de julio de 2024

Acknowledgements

It has been a very intense course, which I am bringing to a close with this work. First of all, I would like to thank the teachers of the master's degree for their work, which despite the precipitous circumstances in which the course has developed, we have managed to survive, to get through it, and above all to learn a lot. I would also like to thank Serena and Miguel Ángel for their willingness to embark on this project and for guiding both Jordi and myself in its development. Of course also to Jordi, for making work not just work, for those meetings that almost always took longer than expected, and for his patience, it is a pleasure to work with people who make everything easier. Finally, thanks to my friends and family for their support, for putting up with my constant complaints about this work and for always being there.

Contents

Contents	IV
List of Figures	V
List of Tables	VII
Abstract	1
Resumen	2
1 Introduction	3
1.1 Criticality and its presence in living systems	3
1.2 Point processes	6
1.2.1 Poisson processes	7
1.2.2 Hawkes processes	8
2 Objectives	12
3 Methodology	13
3.1 Algorithms for time series generation	13
3.2 The importance of time binning	17
4 Results	20
4.1 Results for $n=1$	20
4.2 Results for $n=2$	23
4.3 Results for the bivariate case with excitation and inhibition	26
5 Conclusions	31
A Hawkes processes with exponential kernel	34
B Python scripts	35

List of Figures

1.1	Representation of a power-law distribution. The slope of the double logarithm plot is the exponent of the distribution.	4
1.2	On the left, the dynamics of the SIS (Susceptible-Infected-Susceptible) model, which is a typical contact process for disease spreading [3]. On the right, the order parameter versus the control parameter; in this case, the control parameter would be the infection rate λ . The order parameter is the fraction of infected individuals. The SIS model has a critical point (in a fully-connected network) at $\lambda = \lambda_c = \mu$	4
1.3	On the left, a subcritical branching process, in the middle, a critical branching process and on the right, a supercritical branching process. Each color represents a different generation.	5
1.4	On the left, a subcritical percolation process; on the right, a critical percolation process. The blue line represents a percolant cluster.	6
1.5	Representation of a point process. The intensity function $\lambda(t)$ is a time-dependent function which represents the rate of occurrence of an event.	7
1.6	Left: event number in time for different rates. Right: probability of having a certain number of events for different rates.	8
1.7	On the left, a temporal series of $K = 150$ events of a Hawkes process with $\mu = 0.01$, on the right, a raster plot of the same process.	9
1.8	On the left, a temporal series of $K = 150$ events of a Hawkes process with $\mu = 10$, on the right, a raster plot of the same process.	9
1.9	First, a temporal series of $K = 10^5$ events of a Hawkes process with $\mu = 0.01$, on the right, the event distribution. Thicker blue bars represent avalanches of activity.	10
1.10	First, a temporal series of $K = 10^5$ events of a Hawkes process with $\mu = 10$, on the right, the event distribution.	10
1.11	Scheme of the interaction between the excitatory and inhibitory populations. As illustrated, the excitatory population can excite itself and the inhibitory population, while the inhibitory population can only inhibit the excitatory population, because on most cases the auto-inhibition is negligible [18].	11
3.1	Diagram to calculate the cumulative probability of the inter-event time.	14
3.2	Five a temporal series of $K = 10^5$ events of Hawkes processes with $\mu = 10^{-4}$ on the left side and $\mu = 10^2$ on the right one.	18
3.3	Diagram for $\mu \ll 1$. Red lines represent the events, clusters are coloured. As we can see, we have two regimes, one when Δ is of the order of the average cluster size and another when it is of the order of the inter-event time where the system percolates. . .	19
3.4	Diagram for $\mu \gg 1$. Red lines represent the events, clusters are coloured. In this situation, events occur more regularly, resulting in a unique percolation transition, due to Δ being approximately equal to the inter-event time.	19

4.1	Percolation phase diagrams for different number of events K taking average values of $R = 1000$ realizations. We have included the homogeneous Poisson process at the left for comparison.	20
4.2	Susceptibility χ normalized to the number of events K , for different event number K and taking average values for $R = 1000$ realizations.	21
4.3	Avalanche analysis for Hawkes process with $n = 1$, $K = 10^5$ events. The histograms have been calculated over $R = 1000$ time series.	22
4.4	On the left, the vertical dashed lines represent the critical points $\Delta^*(K)$ for the Poisson process. On the right, the vertical dashed lines represent the critical points $\Delta_1^*(K)$ given by Eq. 4.2 and the dotted dashed lines the $\Delta_1^*/100$	23
4.5	Event series for $n = 1$ and $n = 2$	24
4.6	Percolation phase diagrams for a Hawkes process with $n = 2$ compared to the homogeneous Poisson process on the left.	24
4.7	Susceptibility χ normalized to K , for different number of events and taking average values for $R = 1000$ realizations.	25
4.8	Avalanche statistics for a self-exciting Hawkes process with $n = 2$ for $K = 10^5$ events. The histograms have been calculated over $R = 1000$ time series.	25
4.9	On the left, a temporal series of $K = 10^4$ events of a bivariate Hawkes process with the interaction parameters of the pseudo-critical signal shown in Table 4.1, on the right, the raster plot of the same process for excitatory and inhibitory events.	26
4.10	On the left, a temporal series of $K = 10^4$ events of a bivariate Hawkes process with the interaction parameters of the “stationary” signal shown in Table 4.1, on the right, the raster plot of the same process for excitatory and inhibitory events.	27
4.11	Percolation phase diagrams averaged over $R = 1000$ “pseudo-critical” signals of $K = 10^5$ events. ($n_{EE} = n_{EI} = 1.5, n_{IE} = -0.33$)	27
4.12	Susceptibility χ normalized to the number of events associated with the above phase diagram.	28
4.13	Avalanche statistics of $K = 10^5$ events for “pseudo-critical” signals of two coupled Hawkes processes. Histograms have been calculated over $R = 1000$ time series.	28
4.14	Percolation phase diagrams averaged over $R = 1000$ “stationary” signals of $K = 10^5$ events. ($n_{EE} = n_{EI} = 1.5, n_{IE} = -0.5$)	29
4.15	Susceptibility χ normalized to K associated with the above phase diagram.	29
4.16	Avalanche statistics of $K = 10^5$ events for “stationary” signals of two coupled Hawkes processes. Histograms have been calculated over $R = 1000$ time series.	30

List of Tables

3.1	Configuration of the parameters for the simulations of the article [11]	17
4.1	Interaction parameters for both bivariate processes	26
4.2	Power-law exponents for every configuration studied.	30

List of Algorithms

1	Slow method to generate Hawkes processes.	13
2	Algorithm to generate K Hawkes events.	16
3	Algorithm to generate K Hawkes events for two coupled processes.	16

Abstract

In this master thesis we have explored the relationship between Hawkes processes and criticality in living systems. These processes are an exceptional tool to model self-exciting events, which are relevant in several contexts, such as for the brain, where neurons excite or inhibit each other eliciting cascades of events, in earthquakes, where a seismic event can trigger another one, or in social media, where a post can provoke reactions in the audience. We begin by reviewing stochastic processes, specifically point processes, different examples of them, and where they can be found. Next, in order to understand the dynamics of self-exciting phenomena, we focus on Hawkes processes.

In this thesis, we developed computational methods to generate these processes and analyse their behaviour. We have studied the critical behaviour of a single Hawkes process, the supercritical behaviour of a self-exciting process, and the dynamics of an excitatory and inhibitory coupled Hawkes process. For each of these cases, several sets of parameters have been explored, both reproducing known results and obtaining new valuable insights. These findings give us a better understanding of the criticality underlying these processes and their applications in numerous living systems. Moreover, in the last section of the project we propose future research to shed light on the influence of coupling in the dynamics of the system.

Resumen

En este trabajo fin de máster hemos estudiado la relación entre los procesos de Hawkes y la criticidad en sistemas vivos. Estos procesos son excepcionales para la modelización de eventos autoexcitados, como haber en el cerebro, donde las neuronas pueden excitarse o inhibirse entre sí, en terremotos, donde un seísmo puede provocar otro, o en redes sociales, donde una publicación puede desencadenar reacciones en su audiencia. Este proyecto comienza dando una introducción a los procesos estocásticos, concretamente a los procesos puntuales, presentándose diferentes ejemplos junto a sus aplicaciones. Debido al objetivo principal de la investigación, nos centramos en los procesos de Hawkes, para comprender la dinámica de los fenómenos autoexcitados.

Se han desarrollado métodos computacionales para generar estos procesos y analizar su comportamiento. Se ha estudiado el comportamiento crítico de un único proceso de Hawkes, el comportamiento supercrítico y la dinámica de un proceso de Hawkes excitador acoplado con uno inhibidor. Para cada una de estas configuraciones se han investigado varios conjuntos de parámetros, reproduciendo los resultados ya conocidos y obteniendo nuevos conocimientos para las nuevas configuraciones. Estos descubrimientos nos proporcionan una mejor comprensión de la criticidad que aparece en estos procesos y sus aplicaciones en diferentes sistemas vivos. Además, la última sección abre la posibilidad de futuras investigaciones para comprender la influencia del acople de procesos de Hawkes en la dinámica del sistema.

Chapter 1

Introduction

The following Master's thesis is divided into five chapters. In the first chapter, we will introduce the concept of criticality in complex systems, both living and non-living. We will also make a brief summary of stochastic processes and their relation to point processes and Hawkes processes, which are the main focus of this work. After that, we will present the objectives of this work. In the third chapter, we will present the methodology used in this work, including the simulation of Hawkes processes, and the methodology for its analysis. In the second last chapter, we will present the results of the simulations, including the phase diagram of the process, the susceptibility analysis, and the avalanche analysis and we will discuss them. Finally, in the fifth chapter, we will present the conclusions of this work and the future lines of research.

1.1 Criticality and its presence in living systems

Criticality is one of the key concepts around complex systems because it is believed that they may benefit from by working in a critical state or near it. It has been demonstrated that these states provide the system with desirable properties such as robustness, adaptability, information processing, and a wider dynamic range thanks to the scale invariance associated with the critical states [1]. This scale invariance can be described through a specific probability distribution, named Pareto or power-law distribution like $P(t) = Ct^{-\gamma}$. Where $\gamma \in \text{Re}^+$ and C is a normalization constant. In our case, we will focus on power-law distributions for avalanches of activity. As we will see later, we will be able to define avalanches of activity given a certain temporal window, and we will be able to measure their size (the number of events in an avalanche) and duration (the time between the first and last avalanche event). The exponents for these distributions will be called α and τ for the size and duration of the avalanches, respectively. One way to obtain these exponents is by the slope of the double logarithm plot of the probability distribution, as shown in Figure 1.1. Pareto distribution can be shown to be the only probability distributions that are free of scale. These distributions are observed in many systems, such as earthquakes [2], epidemics [3] or social interactions [4, 5]. Although it is true that there are other mechanisms that can lead to power-law distributions, such as preferential attachment [6], we will focus on criticality.

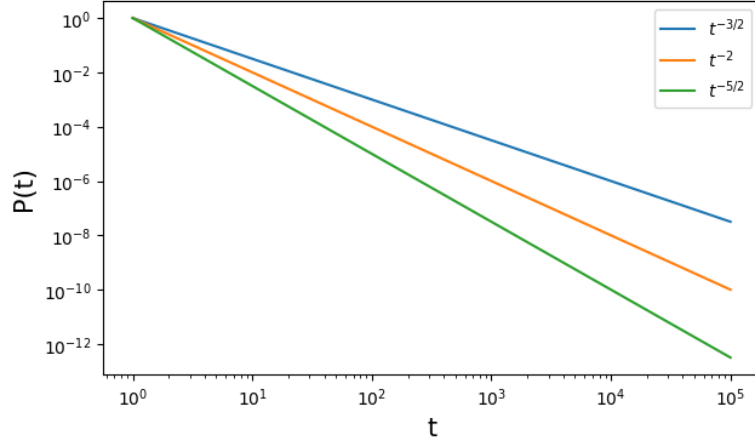


Figure 1.1. Representation of a power-law distribution. The slope of the double logarithm plot is the exponent of the distribution.

The classical toy model to describe criticality is the Ising model [7], which describes the dependence on the temperature of the magnetic properties of metals. The Ising model shows a phase transition at a critical temperature T_c , separating an ordered phase where the magnet's spins are aligned and when the temperature is lower than the critical temperature, and a disordered phase, dominated by thermal fluctuations when $T > T_c$. Ising model is an example of criticality in a system in equilibrium, but these phenomena are also present in systems out of equilibrium. Some of the most studied systems in this context are contact processes due to their simplicity which are used to study activity propagation in a network. The contact processes dynamics are based on the state of the system's constituents, which can be active or inactive, and they can change their state by interacting with each other at a certain rate. For instance, an active site can activate an inactive site with a rate λ which will be our control parameter and the inactive site can deactivate with a rate μ . These rates will be the key to understanding the system's behaviour. In order to analyze the macroscopic properties of the system, we need to define the order parameter, which is a quantity that allows us to establish the phase [8]. An example is shown in Figure 1.2.

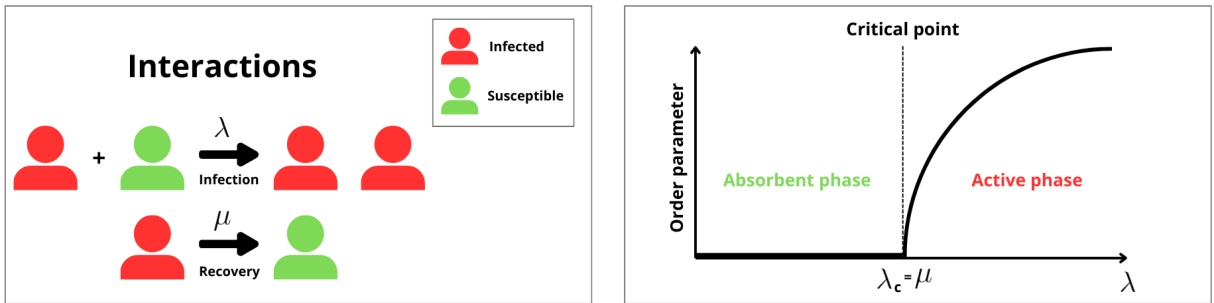


Figure 1.2. On the left, the dynamics of the SIS (Susceptible-Infected-Susceptible) model, which is a typical contact process for disease spreading [3]. On the right, the order parameter versus the control parameter; in this case, the control parameter would be the infection rate λ . The order parameter is the fraction of infected individuals. The SIS model has a critical point (in a fully-connected network) at $\lambda = \lambda_c = \mu$.

Two other classic examples that are related to the results of this work are (critical) branching processes [9] and percolation processes [10]. The first one is used for example to model phenomena related to offspring. In a simplified way we can say that a parent can have a random number of children given by a probability distribution with a mean value of n . We can discern between subcritical, critical and supercritical branching processes depending on this value. For the subcritical case, $n < 1$, and the process will die out eventually because the number of children will decrease on average with each generation. For the supercritical case, $n > 1$, the descendant population will grow exponentially. The most interesting situation is the critical case, $n = 1$, where we will have, in average, power law distributions for the tree size (number of descendants) and the tree depth (number of generations) with exponents $\alpha \sim Z = 3/2$ and $\tau \sim D = 2$ respectively [11]. A representation of these situations is shown in Figure 1.3.

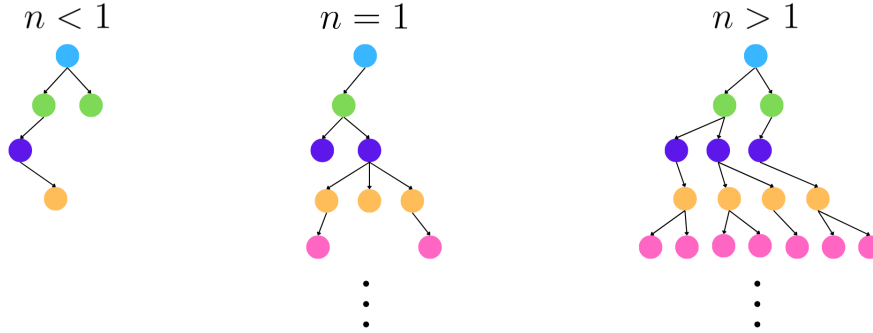


Figure 1.3. On the left, a subcritical branching process, in the middle, a critical branching process and on the right, a supercritical branching process. Each color represents a different generation.

As a last example, we present percolation. These processes are generally used to model the flow of a fluid through a porous medium. Taking into account that this master thesis is related to living systems, there is no better example that relates the model with the topic, and physics (indeed, physicist) than an Italian coffee maker. Here, the porous medium is coffee, and the fluid is water. To describe the percolation process in a simple way, we can say that coffee makes a square lattice, and the water will flow through the nodes with a certain probability p , which depends on the pressure exerted during the pressing process. We can define clusters of connected nodes (in this situation, we are only interested on the vertical connections in order to make coffee) for a given probability, and we also define the order parameter as the size of the largest cluster. Same as with branching processes, we can discern between subcritical, critical and supercritical percolation processes depending on the value of p . For the subcritical case, $p < p_c$, the largest cluster will be smaller than the system size, describing a situation in which the water is not able to flow through the coffee. Conversely, in the supercritical case, $p > p_c$, a lot of cluster will be of the size of the system, describing a situation in which the water flows through the coffee very easily, getting a watery coffee. In the same manner as the branching processes, the most interesting situation is the critical case, $p = p_c$. The critical point describes a situation in which the water can flow through the coffee, but not extremely easily (as would happen in the supercritical case, $p > p_c$) and therefore extracts the coffee aroma in an optimal way. In the critical case, the cluster size, which is the number of clusters given a volume, s , normalized by the total volume, follows a power law distribution $n_s \sim s^{-\tau}$. A representation of these situations is shown in Figure 1.4.

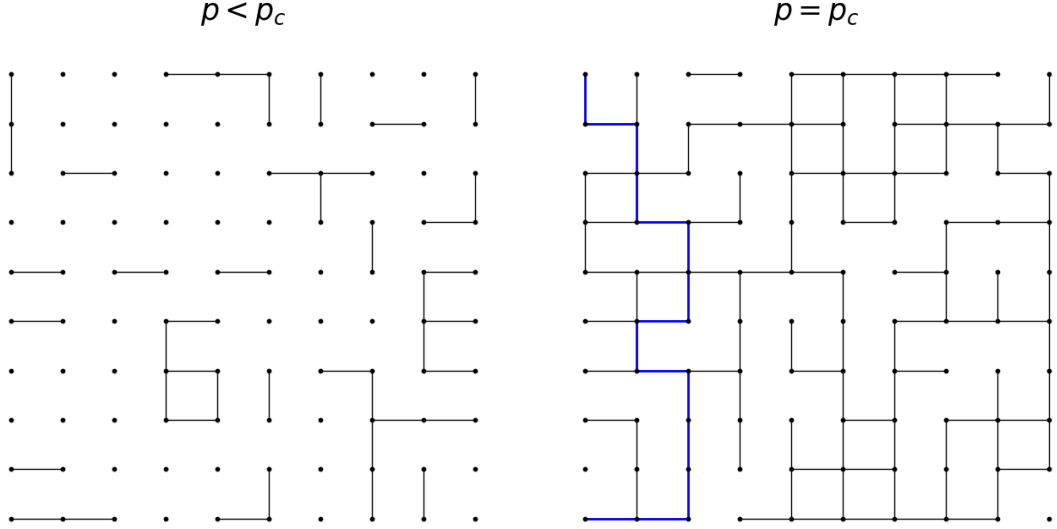


Figure 1.4. On the left, a subcritical percolation process; on the right, a critical percolation process. The blue line represents a percolant cluster.

For our purposes, we will be interested in 1D percolation, whose exponents are $\alpha = 2$ for the cluster size and $\tau = 2$ for the duration [10]. Having seen these examples, we move on to a brief introduction to point processes, which will lead us to another context to discuss criticality.

1.2 Point processes

Within the large framework of complex systems, stochastic processes lend us a hand to decypher properties of living systems, bridging randomness with structured behaviour. These processes are used to model the dynamics of systems that evolve randomly over time. This is why they are ideal for describing natural phenomena such as the spread of diseases [12], social networks [4] or ecological systems [13]. Mathematically, a stochastic process is a collection of random variables [14], generally ordered in time $\{X_t\}_{t \in T}$, where t is the time and X_t is the system state at time t . T is the time index set, which can be discrete or continuous. In this work we will focus on the continuous case because we are interested in the study of point (Hawkes) processes for modeling neurons.

Point processes are a type of stochastic process that describes the occurrence of events in time or space. We will be interested in time point processes because we are going to model the spiking activity of neurons. For our purposes, they will be characterized by two parameters, the time of occurrence of the events t_k and the intensity or rate of occurrence of these events $\lambda(t)$. This rate tells us how likely it is that an event occurs at time t given the history of the process, as pictured in Figure 1.5.

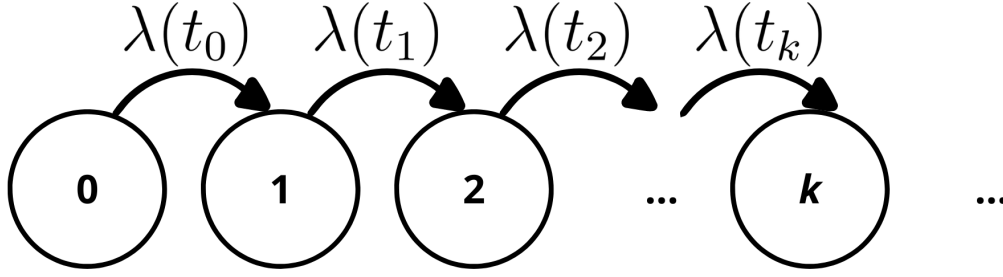


Figure 1.5. Representation of a point process. The intensity function $\lambda(t)$ is a time-dependent function which represents the rate of occurrence of an event.

1.2.1 Poisson processes

In the more general case, the rate is a function of the history of the process, which makes the process non-Markovian, but in our case, it will be a Markovian process, which means that the rate depends only on the last event that occurred, as we will see. An example of a Markovian point process is the Poisson process, which is simple and one of the most studied point processes because it describes events that occur independently in time, such as the arrival of customers at a store or occurrence of defects on a production line. They are also present in some physics phenomena, for instance, the decay of radioactive particles or the arrival of photons at a detector. These processes are characterized by the rate of occurrence of events λ , which is independent of the state of the system. The dynamics of these processes are described by the Poisson distribution, which is the probability distribution that the random variable N takes the value n and describes the probability that n events occur in a given time interval is:

$$P(N = n) = \frac{\lambda^n}{n!} e^{-\lambda}. \quad (1.1)$$

Furthermore, the mean value and the variance of the distribution are equal to λ . Poisson processes can be homogeneous or inhomogeneous, depending on whether the rate is constant or time-dependent. In Figure 1.6 we can see an example of a homogeneous Poisson process.

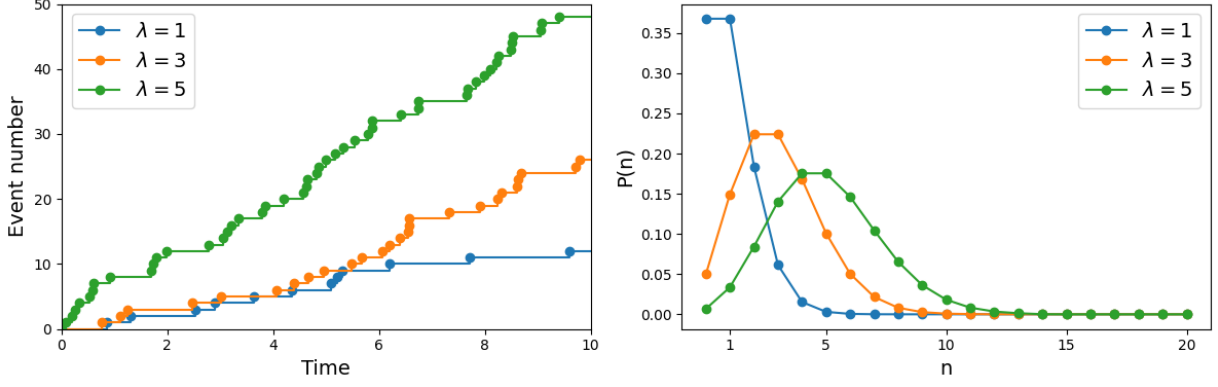


Figure 1.6. Left: event number in time for different rates. Right: probability of having a certain number of events for different rates.

1.2.2 Hawkes processes

On the other hand, if we consider a non-homogeneous Poisson process, the rate is a function of time, $\lambda(t)$. One example of inhomogeneous Poisson processes is in the Hawkes processes, which describes a process that is self-exciting, i.e. the more events happen in a given time window, the more events are likely to happen in the next time window. The rate can be written in several ways [11, 15, 16, 17]. We will use the expression from [11]:

$$\lambda(t|t_1, \dots, t_k) = \mu + n \sum_{i=1}^k \phi(t - t_i), \quad (1.2)$$

where μ is the background rate of a homogeneous Poisson process, n is a parameter that controls the strength of the self-excitation, and $\phi(t)$ is the kernel function that describes the influence of past events on the future rate. The kernel function must be a non-negative and non-monotonically-increasing function that integrates to 1. Typical choices for the kernel function are the exponential or power-law functions, in this work, we will focus on the exponential kernel. Eq. 1.2 models a firing rate that increases by a quantity n with each event, and recovers (guided by the kernel function) toward μ in the absence of any events (see Figure 1.7). We can see that the rate depends on the history of the process, making it non-Markovian in general, but with an exponential kernel, the dependence on the past events decays exponentially fast. In fact, it can be shown that the process becomes Markovian, see Appendix A.

Despite being a Markovian process, it is still an inhomogeneous Poisson process because the rate is not constant. In addition, it is a self-exciting process, which means that the occurrence of an event increases the probability of the occurrence of another event. This is why it is used to model the spiking activity of neurons, where the occurrence of a spike increases the probability of the occurrence of another spike [1]. This self-excitation will enable the appearance of bursts of activity that we will measure. The parameters chosen for the kernel function will be $\alpha = \beta = 1$ and we will vary the background rate μ from values much smaller than 1 to values greater than 1. In Figures 1.7 and 1.8 we can see typical diagrams of Hawkes processes with these parameters. Details on the numerical simulation are explained later on, in Chapter 3.

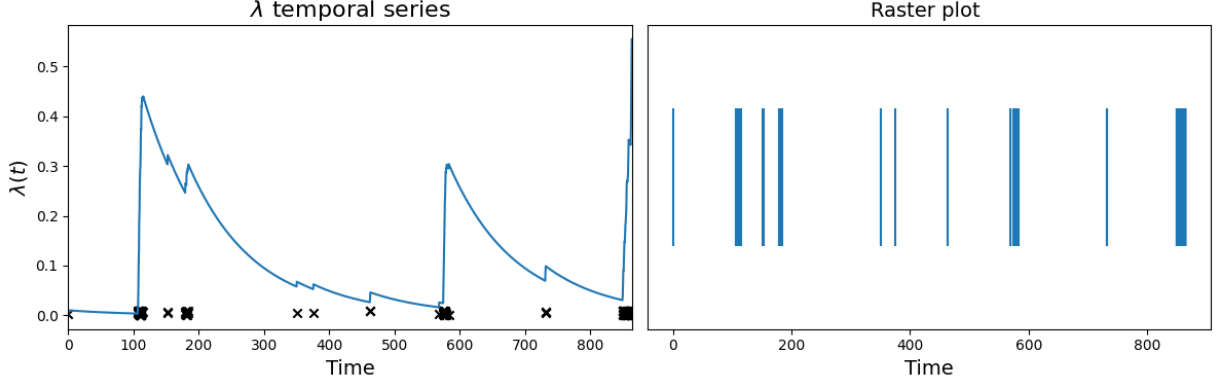


Figure 1.7. On the left, a temporal series of $K = 150$ events of a Hawkes process with $\mu = 0.01$, on the right, a raster plot of the same process.

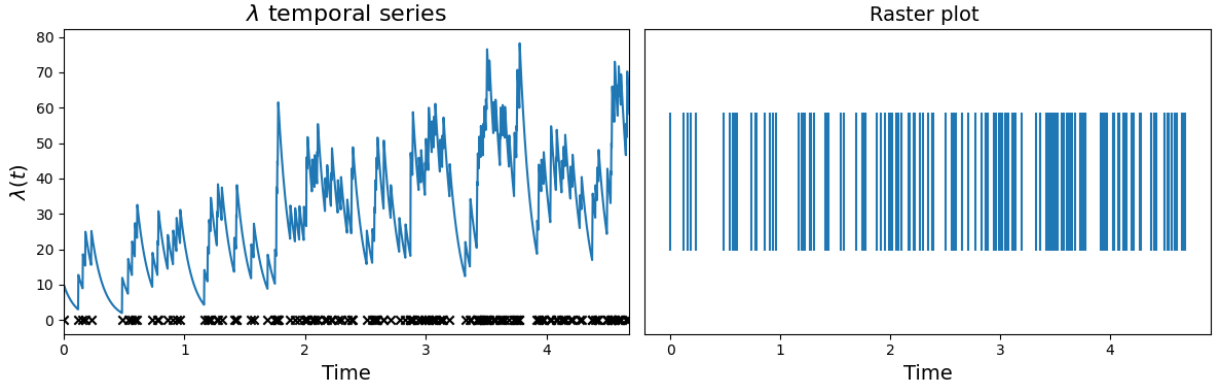


Figure 1.8. On the left, a temporal series of $K = 150$ events of a Hawkes process with $\mu = 10$, on the right, a raster plot of the same process.

As shown in Figure 1.7, when the background rate is smaller than 1, events are less likely to occur, but when they do, they tend to form avalanches of activity thanks to the self-excitation. On the other hand, when the background rate is greater than 1, events occur more frequently, forming avalanches of activity more frequently and longer, as shown in Figure 1.8. If we ignore the time of occurrence of the events and focus only on the structure of λ and therefore of the events, we can see that the process with $\mu = 0.01$ has a bursty structure, while the process with $\mu = 10$ has a more regular structure. This phenomenon is exposed in Figures 1.9 and 1.10.

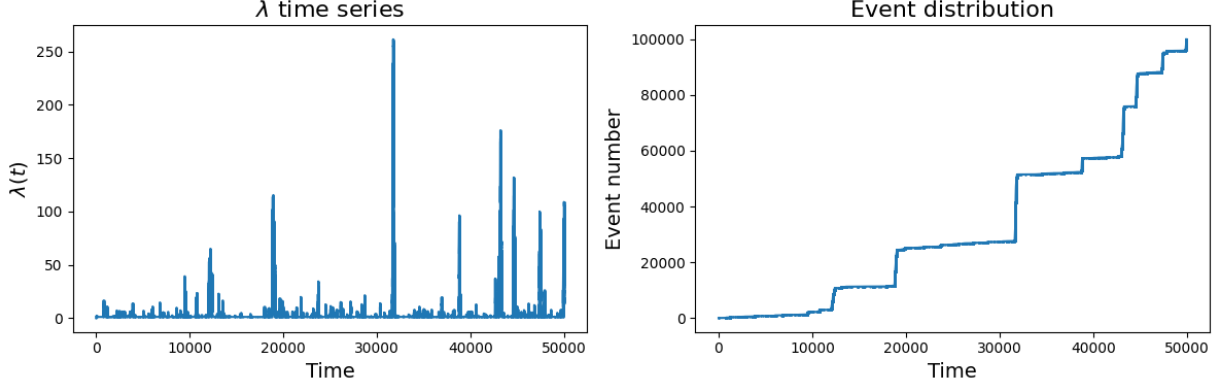


Figure 1.9. First, a temporal series of $K = 10^5$ events of a Hawkes process with $\mu = 0.01$, on the right, the event distribution. Thicker blue bars represent avalanches of activity.

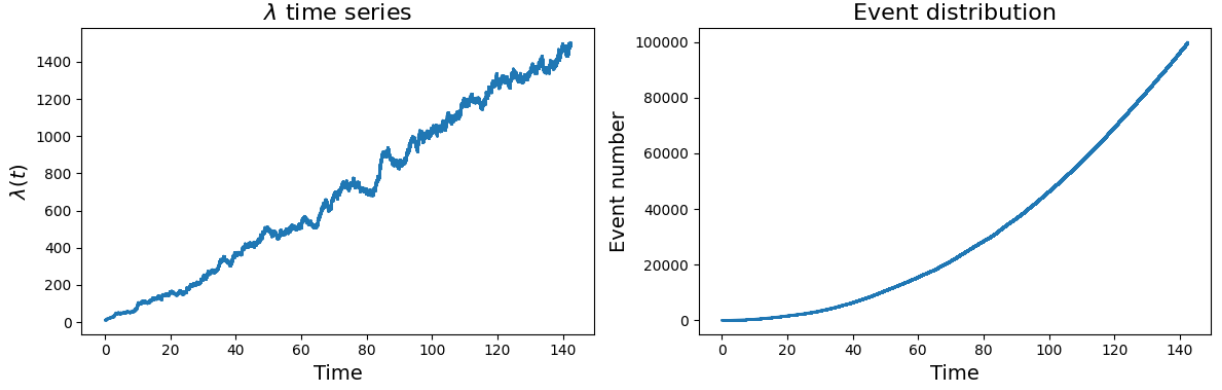


Figure 1.10. First, a temporal series of $K = 10^5$ events of a Hawkes process with $\mu = 10$, on the right, the event distribution.

In most cases, the motivation for studying point processes is counting the events, but in our case, we are also interested in the time of occurrence of the events which will let us define bursts or avalanches of activity that we will use to describe the dynamics of the system. As we have said previously, $\alpha = 1$, $\beta = 1$, making n the parameter that controls the strength of the self-excitation. Essentially, n and α play the same role, so we will use them indistinctly. The previous figures showed the dynamics for $n = 1$, making the system critical as a critical branching process [11], such the one that we previously described. We will also study the dynamics for $n = 2$ and two coupled Hawkes processes.

Bivariate Hawkes processes

With the previous knowledge, we can model an isolated neuron working in different regimes, but as we know, the brain is composed of an enormous number of neurons connected to each other, forming a network. Moreover, neurons can be classified into two kinds, excitatory and inhibitory neurons. To get closer to modeling the brain, we will also generate two coupled Hawkes processes, one corresponding to an excitatory population and the other to an inhibitory population, or just an excitatory and inhibitory neuron. Both populations (neurons) will have a background rate μ_E and μ_I , and they

will be able to interact with each other and with themselves, the “strength” of the interactions will be controlled by the parameters n_{EE} , n_{EI} , n_{IE} and n_{II} . In most cases, the auto-inhibition can be considered negligible. These interactions are illustrated in Figure 1.11 and the equations to model the interaction in Eq. 1.3.

$$\begin{aligned}\lambda_E &= \mu_E + n_{EE} \sum_{i=1}^k \phi(t - t_i) + n_{EI} \sum_{i=1}^k \phi(t - t_i) \\ \lambda_I &= \mu_I + n_{IE} \sum_{i=1}^k \phi(t - t_i) + n_{II} \sum_{i=1}^k \phi(t - t_i)\end{aligned}\tag{1.3}$$

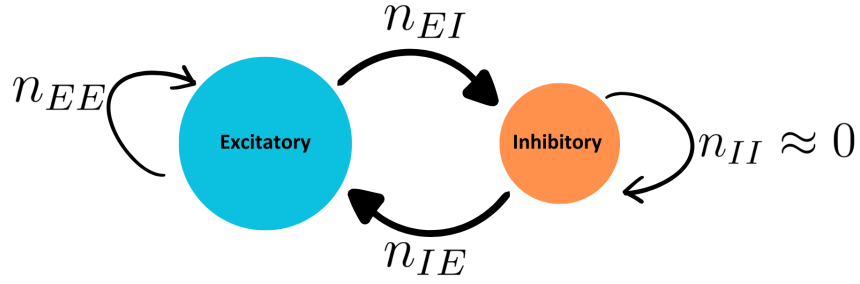


Figure 1.11. Scheme of the interaction between the excitatory and inhibitory populations. As illustrated, the excitatory population can excite itself and the inhibitory population, while the inhibitory population can only inhibit the excitatory population, because on most cases the auto-inhibition is negligible [18].

Chapter 2

Objectives

The main objectives of this master's thesis are:

- To understand the basics of criticality and its appearance in living systems, particularly in the brain.
- To understand what Hawkes processes are, where we can find them, how to generate them computationally and relate them with neuroscience.
- To analyse the effect of parameters specific to the analysis temporal of the analysis (e.g. temporal binning in raster plots).
- To reproduce the results of the original paper [11] and compare them with the results obtained in this work.
- To study the behaviour of a supercritical self-exciting process and compare it with the results from a critical process.
- To study how an excitatory and inhibitory coupled Hawkes process behaves.

Chapter 3

Methodology

In the following sections, the methodology for data generation, management and analysis will be presented. To address these issues, we will use Python [19, 20] due to its versatility and the wide range of libraries available. The two used will be NumPy [21] and Matplotlib [22] for the visualization.

3.1 Algorithms for time series generation

The first step is the generation of time series. There are two ways to do this: the slow one and the fast one. The first one is discretizing the time and calculating the rate at each time step according to Eq. 1.2, then accept or reject the event if $p < \lambda \cdot dt$ for a random number $p \in \mathcal{U}[0, 1]$. This method works for small time series, but for large ones is not efficient because the summation of the kernel function has to be done at each time step. The pseudo-code for this method is presented in Algorithm 1.

Algorithm 1 Slow method to generate Hawkes processes.

Require: t_{\max} , $n_{\text{intervals}}$, $\lambda(t_0) = \mu$, p

$$dt \leftarrow \frac{t_{\max}}{n_{\text{intervals}}}$$

for $i = 0$ to $n_{\text{intervals}}$ **do**

$$\lambda(t_k) \leftarrow \mu + n \sum_{t_i < t_k} \phi(t_k - t_i)$$

$$\triangleright t_i = i \cdot dt$$

if $\lambda(t_k) \cdot dt > p$ **then**

$$t_{\text{event}} \leftarrow t_k$$

end if

end for

The fast method takes advantage of Monte Carlo methods [23] to generate the time series. The idea behind this procedure is to compute the inter-event time instead of the time of the event. To get to the algorithm, we start from the following expression:

$$PDF(\text{inter-event time} = \Delta t) = \lambda(t + \Delta t) e^{-\int_t^{t+\Delta t} \lambda(t') dt'} \quad (3.1)$$

To demonstrate this, we have to take a look at Figure 3.1 and recall that λ is a probability per unit of time.

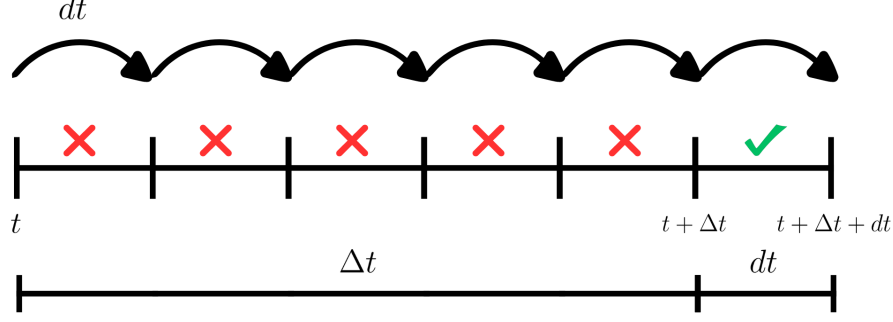


Figure 3.1. Diagram to calculate the cumulative probability of the inter-event time.

The probability that the inter-event interval is Δt is equal to the probability that no events occur in the interval $[t, t + \Delta t]$ times the probability that an event occur in the interval $[t + \Delta t, t + \Delta t + dt]$. Putting words into mathematics, we have that the probability of not having an event in the interval $[t, t + \Delta t]$ is:

$$\begin{aligned}
 P(\text{No event} \in [t + \Delta t + dt]) &= (1 - \lambda(0) \cdot dt) (1 - \lambda(dt) \cdot dt) (1 - \lambda(2dt) \cdot dt) \dots \\
 &= \prod_{k=0}^{\infty} \underbrace{(1 - \lambda(kdt) \cdot dt)}_{e^{\ln(1 - \lambda(kdt)dt)}} = e^{\sum_{k=0}^{\infty} \ln(1 - \lambda(kdt)dt)} = \dots \quad \text{Using } \ln(1 - \varepsilon) \approx -\varepsilon \\
 &= e^{-\sum_{k=0}^{\infty} \lambda(kdt)dt} \underset{dt \rightarrow 0}{=} e^{-\int_t^{t+\Delta t} \lambda(t')dt'}.
 \end{aligned} \tag{3.2}$$

Knowing Eq. 3.1 we can calculate the inter-event time following the next steps. In order to generate Δt , we will use the inverse transform method [24], therefore, we have to calculate the cumulative probability of the inter-event time:

$$\begin{aligned}
 \text{accum}(\Delta t) &= \int_0^{\Delta t} \text{PDF}(\Delta t') d\Delta t' = u \in \mathcal{U}[0, 1] \\
 &= \int_0^{\Delta t} \underbrace{\lambda(t + \Delta t') e^{-\int_t^{t+\Delta t'} \lambda(t')dt'}}_{-\frac{d}{d\Delta t'} \left[e^{-\int_t^{t+\Delta t'} \lambda(t')dt'} \right]} d\Delta t' = u \quad \text{Using Barrow rule} \\
 &= e^{-\int_t^{t+\Delta t'} \lambda(t')dt'} \Big|_0^{\Delta t} = 1 - e^{-\int_t^{t+\Delta t} \lambda(t')dt'} = u \quad \text{Taking logarithms} \\
 &\int_t^{t+\Delta t} \lambda(t')dt' = -\ln(1 - u) = -\ln(\bar{u})
 \end{aligned} \tag{3.3}$$

To compute the inter-event time, we have to generate $\bar{u} \sim \mathcal{U}[0, 1]$ and solve the equation. Having in mind this relation and using Eq. A.2 we have:

$$\begin{aligned}
 u &= 1 - e^{-\mu(t-t_k)} e^{-\overbrace{(\lambda(t_k) + \alpha - \mu) \cdot \int_{t_k}^t e^{-\beta(t'-t_k)} dt'}^{-\frac{1}{\beta} [e^{-\beta(t-t_k)} - 1]}} \\
 u &= 1 - \underbrace{e^{-\mu(t-t_k)}}_{P(t_{k+1}^{(1)} > t)} e^{-\underbrace{[(\lambda(t_k) + \alpha - \mu)\beta^{-1} (1 - e^{-\beta(t-t_k)})]}_{P(t_{k+1}^{(2)} > t)}}
 \end{aligned} \tag{3.4}$$

Where we have interpreted these two factors as probabilities (because they are smaller than one) of the inter-event time being greater than t . This is the key to generate the inter-event time because if we do not decompose them, we can not invert the function. Then we apply the composition method [16]. If we take $t_{k+1} = \min(t_{k+1}^{(1)}, t_{k+1}^{(2)})$; then $t_{k+1} \sim P(t_{k+1} > t)$, hence:

$$\begin{aligned} \text{Prob}\left(t_{k+1} = \min\left(t_{k+1}^{(1)}, t_{k+1}^{(2)}\right) \leq t\right) &= 1 - \text{Prob}\left(\min\left(t_{k+1}^{(1)}, t_{k+1}^{(2)}\right) > t\right) \\ &= 1 - \text{Prob}\left(t_{k+1}^{(1)} > t\right) \cdot \text{Prob}\left(t_{k+1}^{(2)} > t\right) \end{aligned} \quad (3.5)$$

where we have used that the probability that the smaller is greater than t is that each separately is greater than t . As we can see, the expressions in Eqs. 3.4 and 3.5 are the same, so we can use the composition method to generate the inter-event time. Then, the algorithm to generate the inter-event time is:

1. Generate $t_{k+1}^{(1)} \sim P\left(t_{k+1}^{(1)} > t\right) = e^{-\mu(t-t_k)}$ using

$$P\left(t_{k+1}^{(1)} \leq t\right) = 1 - \underbrace{e^{-\mu(t-t_k)}}_{\substack{\bar{u}_1 \in \mathcal{U}[0,1] \\ 1-\bar{u}_1=u_1}} = u_1 \in \mathcal{U}[0,1]$$

This is done by generating $u_1 \in \mathcal{U}[0,1]$ and solving the equation.

$$\begin{aligned} u_1 &= 1 - e^{-\mu(t_{k+1}^{(1)}-t_k)} \\ \ln(u_1) &= -\mu(t_{k+1}^{(1)}-t_k) \Rightarrow t_{k+1}^{(1)} = t_k - \frac{\ln(u_1)}{\mu} \end{aligned} \quad (3.6)$$

2. Generate $t_{k+1}^{(2)} \sim P\left(t_{k+1}^{(2)} > t\right) = e^{-\left((\lambda(t_k)+\alpha-\mu)\beta^{-1}\left(1-e^{-\beta(t_{k+1}^{(2)}-t_k)}\right)\right)}$ in a similar way as before:

$$\begin{aligned} u_2 &= 1 - e^{-\left((\lambda(t_k)+\alpha-\mu)\beta^{-1}\left(1-e^{-\beta(t_{k+1}^{(2)}-t_k)}\right)\right)} \\ -\ln(u_2) &= \left((\lambda(t_k)+\alpha-\mu)\beta^{-1}\left(1-e^{-\beta(t_{k+1}^{(2)}-t_k)}\right)\right) \\ 1 + \frac{\beta \ln u_2}{\lambda(t_k) + \alpha - \mu} &= e^{-\beta(t_{k+1}^{(2)}-t_k)} \\ t_{k+1}^{(2)} &= t_k - \beta^{-1} \ln \underbrace{\left(1 + \frac{\beta \ln u_2}{\lambda(t_k) + \alpha - \mu}\right)}_{\text{This number must be positive}} \end{aligned} \quad (3.7)$$

3. Choose $t_{k+1} = \min\left(t_{k+1}^{(1)}, t_{k+1}^{(2)}\right)$

4. Calculate the rate at t_{k+1} using Eq. A.1 and go back to step 1.

With this method, we can generate time series efficiently. The pseudo-code for this method is presented in Algorithm 2.

To conclude the time series generation section, we can generalize the algorithm in order to generate M coupled Hawkes processes [16, 17]. The essence of the algorithm is the same as the one presented. First, we generate the inter-event time for the excitatory population and the inhibitory population, after that, we choose the minimum of both and update the rates of both populations according to the event that has just occurred. Mathematically, it is expressed as follows:

Algorithm 2 Algorithm to generate K Hawkes events.

Require: $\alpha, \beta, \lambda(t_0) = \mu, K$

for $k = 0$ to K **do**

$u_1, u_2 \leftarrow \mathcal{U}[0, 1]$

$t_{k+1}^{(1)} \leftarrow \frac{\ln(u_1)}{\mu}$

$t_{k+1}^{(2)} \leftarrow \beta^{-1} \ln \left(1 + \frac{\beta \ln u_2}{\lambda(t_k) + \alpha - \mu} \right)$

$t_{k+1} \leftarrow \min(t_{k+1}^{(1)}, t_{k+1}^{(2)})$

$\lambda(t_{k+1}) \leftarrow \mu + e^{-\beta(t_{k+1}-t_k)} (\lambda(t_k) - \mu + n)$

end for

1. Generate $\Delta_{k+1} = \min \{ \Delta_{k+1}^{(1)}, \Delta_{k+1}^{(2)} \}$ with $\Delta_{k+1}^{(j)} = t_{k+1}^{(j)} - t_k^{(j)}$ generated as in Eqs. 3.6 and 3.7.

$$\Delta_{k+1}^{(j)} = \min \left\{ -\frac{\ln(u_1^{(j)})}{\mu_j}, -\beta_j^{-1} \ln \left(1 + \frac{\beta_j \ln u_2^{(j)}}{\underbrace{\lambda_j(t_k^{(j)}) + \alpha_j - \mu_j}_{g_j}} \right) \right\} \quad (3.8)$$

Note that g_j must be positive, otherwise, take the other term.

2. Once we have the process (l), we update the time for the following event as $t_{k+1} = t_k + \Delta_{k+1}^{(l)}$.
3. Update the rates for the excitatory and inhibitory populations as follows:

$$\lambda_j(t_{k+1}) = \mu_j + e^{-\beta_j(t_{k+1}-t_k)} (\lambda_j(t_k) - \mu_j + \alpha_{l \rightarrow j}) \quad \text{with } j = 1, 2 \quad (3.9)$$

The pseudo-code for this method is presented in Algorithm 3.

Algorithm 3 Algorithm to generate K Hawkes events for two coupled processes.

Require: $\alpha_{11}, \alpha_{12}, \beta_1, \mu_1, \alpha_{22}, \alpha_{21}, \beta_2, \mu_2, K$

for $k = 0$ to K **do**

$u_1^{(1)}, u_2^{(1)}, u_1^{(2)}, u_2^{(2)} \leftarrow \mathcal{U}[0, 1]$

$\Delta_{k+1}^{(1)} \leftarrow \min \left(-\frac{\ln(u_1^{(1)})}{\mu_1}, -\beta_1^{-1} \ln \left(1 + \frac{\beta_1 \ln u_2^{(1)}}{\lambda_1(t_k) + \alpha_{11} - \mu_1} \right) \right)$

$\Delta_{k+1}^{(2)} \leftarrow \min \left(-\frac{\ln(u_1^{(2)})}{\mu_2}, -\beta_2^{-1} \ln \left(1 + \frac{\beta_2 \ln u_2^{(2)}}{\lambda_2(t_k) + \alpha_{22} - \mu_2} \right) \right)$

$l \leftarrow \arg \min(\Delta_{k+1}^{(1)}, \Delta_{k+1}^{(2)})$

$t_{k+1} \leftarrow t_k + \Delta_{k+1}^{(l)}$

$\lambda_1(t_{k+1}) \leftarrow \mu_1 + e^{-\beta_1(t_{k+1}-t_k)} (\lambda_1(t_k) - \mu_1 + \alpha_{l \rightarrow 1})$

$\lambda_2(t_{k+1}) \leftarrow \mu_2 + e^{-\beta_2(t_{k+1}-t_k)} (\lambda_2(t_k) - \mu_2 + \alpha_{l \rightarrow 2})$

end for

3.2 The importance of time binning

Now we have a method to generate the main ingredient, time series. In order to cook (analyze) them, our tools will be Python libraries and a control parameter, which in our case will be a resolution parameter $\Delta > 0$ that will allow us to identify clusters of activity. Assuming that we have a time series with K events that happen in times $\{t_1, \dots, t_K\}$. Each event starts a cluster, and the following event will be part of this cluster if the time between both events is less than Δ and so on for the rest (see Figure 3.3). We define the size of the cluster as the number of events in the interval $[t_{first}, t_{last}]$ and its duration as $t_{last} - t_{first}$. The extreme cases are when Δ is smaller than the minimum inter-event time, where each event is a cluster of size 1 and duration 0. The other extreme is when Δ is greater than the largest inter-event time, where all the events are in the same cluster of size K and duration $t_K - t_1$. Between these two extremes, we will have different regimes of the process. Our recipe will be the phase diagram, specifically the percolation diagram, where we will plot the percolation strength P_∞ as a function of the resolution parameter Δ . The percolation strength is defined as the fraction of events that are in the largest cluster over the total number of events, $P_\infty = S_M/K$ where S_M is the size of the largest cluster. Three different sets of parameters will be used to generate the time series in order to compare them. The parameters α and β will be fixed to unless otherwise stated, the other parameters are shown in Table 3.1. Once we got our recipe (the percolation diagram), we should be able to identify the critical points and the different regimes of the process. By carefully choosing Δ from the percolation diagram, we will be able to identify the different regimes with their respective power law exponents.

Table 3.1. Configuration of the parameters for the simulations of the article [11].

Configuration	μ	n
First	1	0
Second	10^{-4}	1
Third	10^2	1

The percolation diagram will be generated by generating 1000 time series for each configuration and calculating the percolation strength for each one because they are stochastic processes, as we can observe in Figure 3.2.

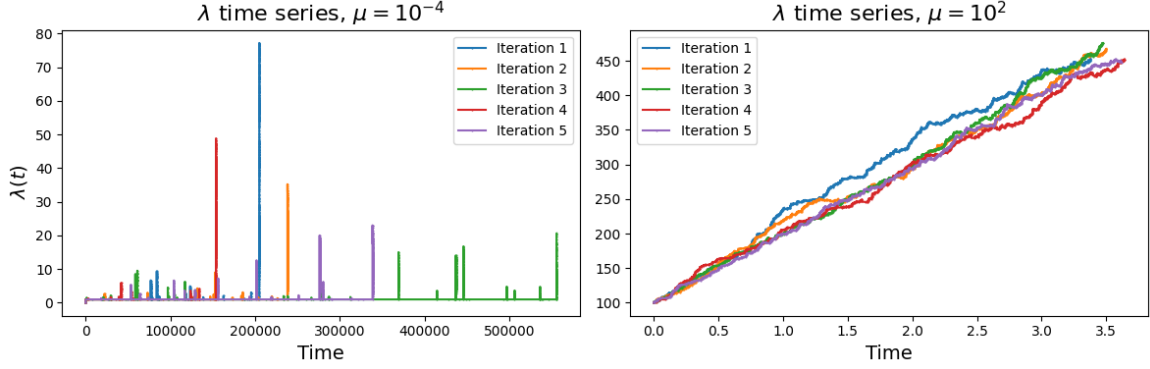


Figure 3.2. Five a temporal series of $K = 10^5$ events of Hawkes processes with $\mu = 10^{-4}$ on the left side and $\mu = 10^2$ on the right one.

Beginning with the first configuration, we have a homogeneous Poisson process, and the inter-event probability of having an inter-event time x_i is given by $P(x_i) = \mu e^{-\mu x_i}$. Consequently, two consecutive events will be part of a cluster, fixing the resolution parameter to Δ with a probability of

$$P(x_i \leq \Delta) = 1 - e^{-\mu \Delta} \quad \forall i. \quad (3.10)$$

This represents the probability in a homogeneous 1D percolation model [10], where we can identify a non percolant phase and a percolant phase separated by the critical point Δ^* . We can calculate this parameter if we know the maximum inter-event time of the time series. Let us assume that our time series has K events. Therefore, it will percolate if the condition we have just established is satisfied. We can calculate this threshold as the average of the maximum inter-event time in K samples from the inter-event time distribution solving the following equation:

$$\begin{aligned} K \int_{\Delta^*}^{\infty} P(x) dx &= 1 \\ K \int_{\Delta^*}^{\infty} \mu e^{-\mu x} dx &= 1 \\ -K [e^{-\mu x}]_{\Delta^*}^{\infty} &= K \left[e^{-\mu \Delta^*} - e^{-\mu \infty} \right] = 1 \\ K e^{-\mu \Delta^*} &= 1 \\ \Delta^*(K) &= -\frac{\ln(K)}{\mu} \end{aligned} \quad (3.11)$$

For the other two configurations, on both we have a self-exciting process with $n = 1$, which means that we have a critical dynamical regime, as we have seen, but with different background rates, one much smaller than 1 and the other much greater than 1. This fact will be reflected in the percolation diagram. We will not approach these cases from a theoretical point of view but from a graphical one. With the second configuration, as we have seen in Figure 1.9 if the condition $\mu \ll 1$ is satisfied, we will have a bursty structure in the time series. Due to the low background rate, the events are less likely to occur, but when they do, they tend to form avalanches of activity thanks to the self-excitation. This will be reflected in the percolation diagram as a first phase transition at a critical point Δ_1^* when Δ is of the order of the average cluster size. Then, a second phase transition will occur at a critical point Δ_2^* when Δ is greater than the greatest inter-event time. This phenomenon is illustrated in Figure 3.3. Note that this very specific analysis is due to the way that activity is measured in the experiments,

where a voltage is measured and the spikes are detected when the voltage is above a certain voltage and temporal threshold.

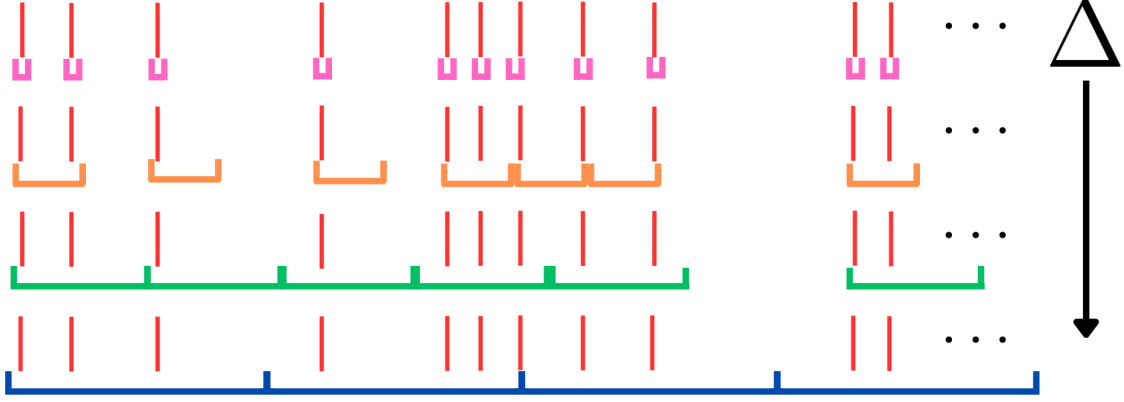


Figure 3.3. Diagram for $\mu \ll 1$. Red lines represent the events, clusters are coloured. As we can see, we have two regimes, one when Δ is of the order of the average cluster size and another when it is of the order of the inter-event time where the system percolates.

On the other hand, when $\mu \gg 1$ events occur more frequently, without making the bursty structure of Figure 1.9, but making a more regular structure as illustrated in Figure 1.10. This will be reflected in the phase diagram as a single phase transition at a critical point Δ^* when Δ is of the order of the average cluster size. This phenomenon is illustrated in Figure 3.4. Note the absence of a time scale in both diagrams, they are examples for the explanation of the phase diagram.

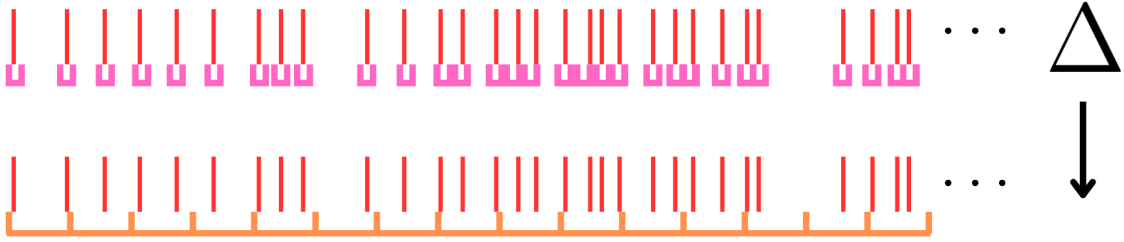


Figure 3.4. Diagram for $\mu \gg 1$. Red lines represent the events, clusters are coloured. In this situation, events occur more regularly, resulting in a unique percolation transition, due to Δ being approximately equal to the inter-event time.

Chapter 4

Results

This section provides the main results of the investigation. The main functions used to obtain these results are shown in the Appendix B. The results are divided into three sections:

1. First, the results reproduced from the original paper [11] are presented.
2. Secondly, we will present the same analysis for the case of a Hawkes process with $n = 2$.
3. Finally, we will study the behaviour of two coupled Hawkes process, one representing an excitatory neuron and the other an inhibitory neuron.

4.1 Results for $n=1$

The structure for the three sections will be the same. First, we will obtain the phase diagram for the percolation strength P_∞ versus our control parameter, the resolution parameter Δ , obtaining the critical point(s) Δ_i^* . Next, the avalanche statistics for size and duration will be studied for different regions of the phase diagram.

As previously stated, the first result is the percolation phase diagram, shown in Figure 4.1 versus the resolution parameter Δ . To obtain this plot, we generated several time series, computed the percolation strength for each one, and plotted the average across realizations.

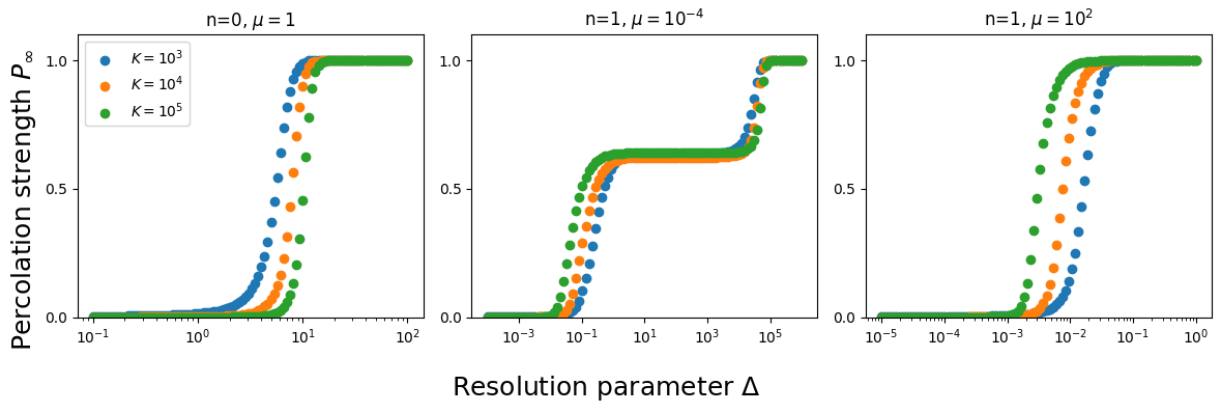


Figure 4.1. Percolation phase diagrams for different number of events K taking average values of $R = 1000$ realizations. We have included the homogeneous Poisson process at the left for comparison.

The first plot configuration is a homogeneous Poisson process with rate $\mu = 1$ which we have overviewed in Section 1.2.1 and has a pseudocritical threshold at $\Delta^*(K) = \frac{\ln(K)}{\mu}$ as we have demonstrated in Section 3.2. Due to the finite size of the time series, the transition is located at the threshold, as expected for 1D percolation [10].

Now, we will consider Hawkes processes, for the first case ($\mu = 10^{-4}$), we can observe a double transition. The first one is at Δ_1^* and the second one at Δ_2^* . As we are going to see with the avalanche statistics, the first transition is associated with the universality class of 1D percolation, whose exponents are $\alpha = \tau = 2$. On the other hand, the second transition is associated with the universality class of the mean-field branching process, whose exponents are $\alpha = 3/2$ and $\tau = 2$. This double transition is also compatible with what was mentioned in Figure 3.3. We can also observe that the plateau between the two transitions is wider as the K increases as expected. For the second case ($\mu = 10^2$), similarly to the first one, we have a single transition at Δ_1^* associated with the universality class of 1D percolation as well; this phenomenon is also compatible with what was shown in Figure 3.4.

Another interesting analysis to characterize the phases is studying the susceptibility χ . Let S_M be the size of the largest cluster, then the susceptibility is defined by Eq. 4.1.

$$\begin{aligned}\chi &= \frac{\langle S_M^2 \rangle - \langle S_M \rangle^2}{\langle S_M \rangle} \\ &= K \cdot \frac{\langle P_\infty^2 \rangle - \langle P_\infty \rangle^2}{\langle P_\infty \rangle} \\ &= K \cdot \frac{\sigma^2(P_\infty)}{\langle P_\infty \rangle}\end{aligned}\tag{4.1}$$

The susceptibility (normalized to the number of events) is shown in Figure 4.2. For the Poisson process, we see that the susceptibility has a peak at the threshold $\Delta^*(K)$, then it vanishes as expected. For the Hawkes case with $\mu = 10^{-4}$, we observe that χ has a peak at the critical point Δ_1^* , then we have a critical behaviour where the susceptibility is not zero at the plateau $[\Delta_1^*, \Delta_2^*]$ and finally, it vanishes at the second critical point, Δ_2^* . Finally, for the Hawkes case with $\mu = 10^2$ and likewise the Poisson process, the susceptibility has a divergence at the critical point Δ_1^* and then it vanishes.

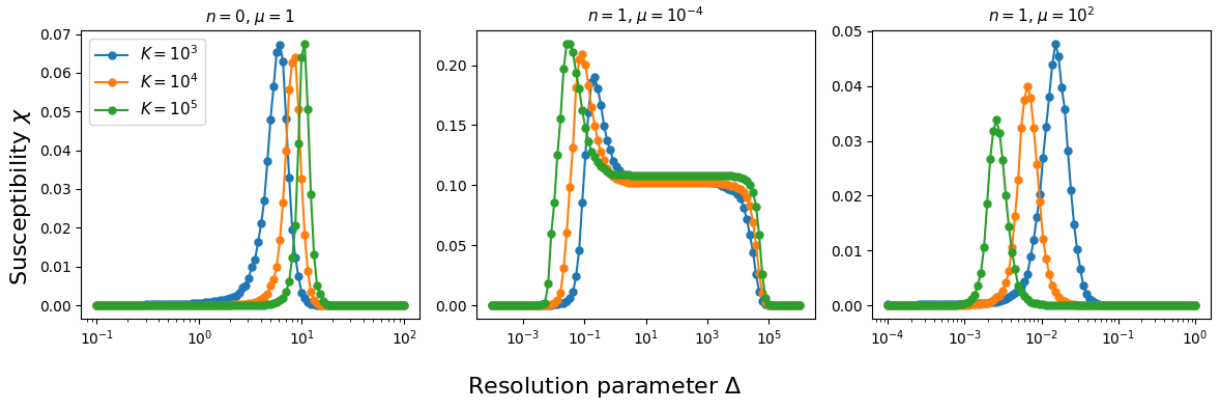


Figure 4.2. Susceptibility χ normalized to the number of events K , for different event number K and taking average values for $R = 1000$ realizations.

Once we have the phase diagram, we can study avalanche statistics, but first, we need to obtain the thresholds Δ_1^* and Δ_2^* from the phase diagram. The article [11] provides the following formulas to compute these thresholds for the Hawkes process with $\mu = 10^{-4}$:

$$\Delta_1^* \simeq \frac{\ln(K)}{\langle \lambda \rangle} = \frac{\ln(K)}{\mu + \sqrt{2\mu K}} \quad (4.2)$$

$$\Delta_2^* = \frac{\ln(K)}{\mu} \quad (4.3)$$

and for $\mu = 10^2$:

$$\Delta_1^* = \frac{\ln(K)}{\mu} \quad (4.4)$$

Bearing this in mind and the definitions of the size and duration of avalanches established in the previous chapter, we can study the avalanches for the different regions of the phase diagram. We are just going to show the results for $\mu = 10^{-4}$ and for $\mu = 10^2$ in Figure 4.3. The Poisson process behaviour can be found in [10, 25].

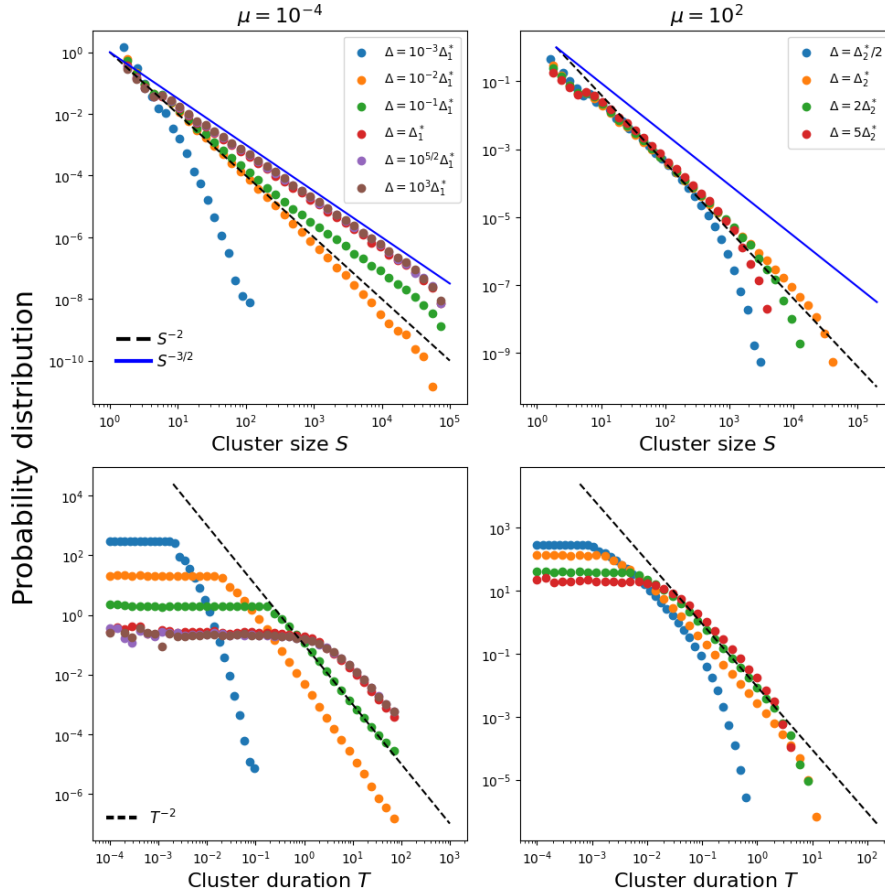


Figure 4.3. Avalanche analysis for Hawkes process with $n = 1$, $K = 10^5$ events. The histograms have been calculated over $R = 1000$ time series.

As a consequence of the huge simulation time for time series with $K = 10^8$ events, we only have studied avalanches for $K = 10^5$, moreover, we have taken other criteria to obtain the histograms. Instead of considering $C = 10^7$ clusters, we have obtained the histograms of $R = 1000$ time series. This leads to a different amount of clusters for each value of Δ . Nevertheless, we have obtained equivalent and robust results.

For $\mu = 10^{-4}$, the probability distribution of the cluster size and duration shows three different behaviours. For $\Delta \ll \Delta_1^*$, the behaviour is subcritical, leading to an exponential decay for the size and duration. As we increase Δ , we reach the critical point where the exponents are $\alpha = \tau = 2$ compatible with the universality class of 1D percolation. After that, we reach the plateau $[\Delta_1^*, \Delta_2^*]$ where we have a crossover to the universality class of mean-field branching process and 1D percolation. Finally, for $\Delta \rightarrow \Delta_2^*$, we obtain the universality class of mean-field branching process exponents $\alpha = 3/2$ and $\tau = 2$. For $\mu = 10^2$, the plots show a power-law distribution for both cluster size and duration with exponents $\alpha = \tau = 2$ corresponding to the universality class of 1D percolation as we have mentioned before.

Note that we have reproduced the same behaviour, but for other values of Δ , specifically, for two orders of magnitude less than in ref. [11]. This is due to the fact that Eq. 4.2 needs the assumption of large time series, which is not fulfilled in our case. We can illustrate this difference, for example, in the susceptibility diagram, where the peak should be at Δ_1^* , but in our case, it is at $\Delta_1^*/100$ as shown in Figure 4.4.

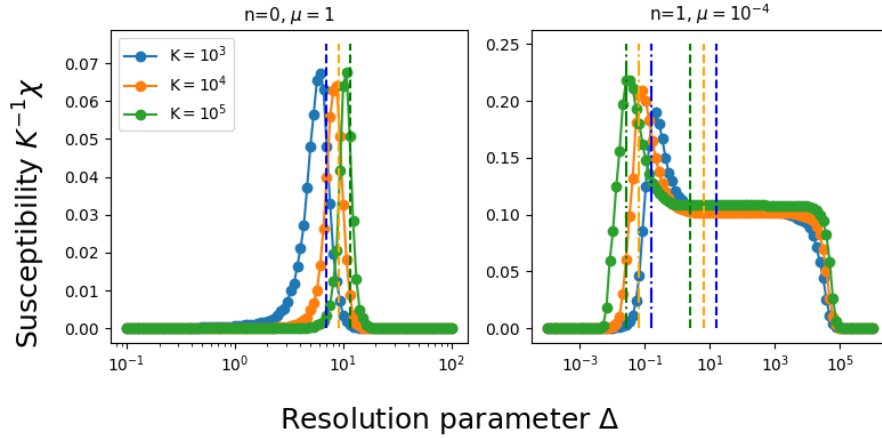


Figure 4.4. On the left, the vertical dashed lines represent the critical points $\Delta_1^*(K)$ for the Poisson process. On the right, the vertical dashed lines represent the critical points $\Delta_1^*(K)$ given by Eq. 4.2 and the dotted dashed lines the $\Delta_1^*/100$.

4.2 Results for $n=2$

In the article, the authors have studied a process that is critical itself because the parameter n is fixed to $n = 1$. We have studied the case $n = 2$ to see if the process is still critical. In Figure 4.5 two event series for $n = 1$ and $n = 2$ are shown.

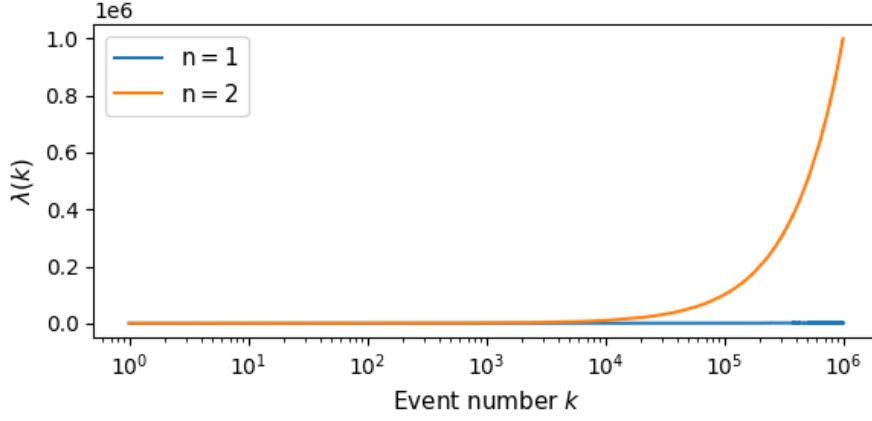


Figure 4.5. Event series for $n = 1$ and $n = 2$.

As presented in the figure above, the rate of the process for $n = 2$ explodes in comparison with the rate for $n = 1$. Similarly to the previous section, the first step is obtaining the phase diagram in order to distinguish the regimes.

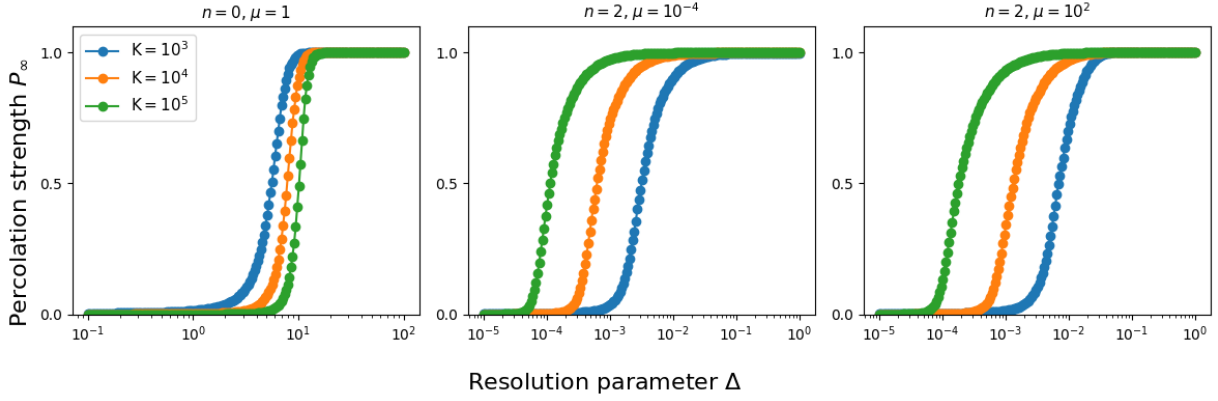


Figure 4.6. Percolation phase diagrams for a Hawkes process with $n = 2$ compared to the homogeneous Poisson process on the left.

In this case, Eqs. 4.2, 4.3 are not valid because they were derived for $n = 1$. Therefore, we will obtain this parameter graphically from the phase diagrams shown in Figure 4.6. We will establish Δ^* at the resolution parameter where the percolation strength $P_\infty = 0.5$, consequently, $\Delta^* \approx 10^{-4}$ for both cases. Like in the previous section, in Figure 4.7 the susceptibility is shown.

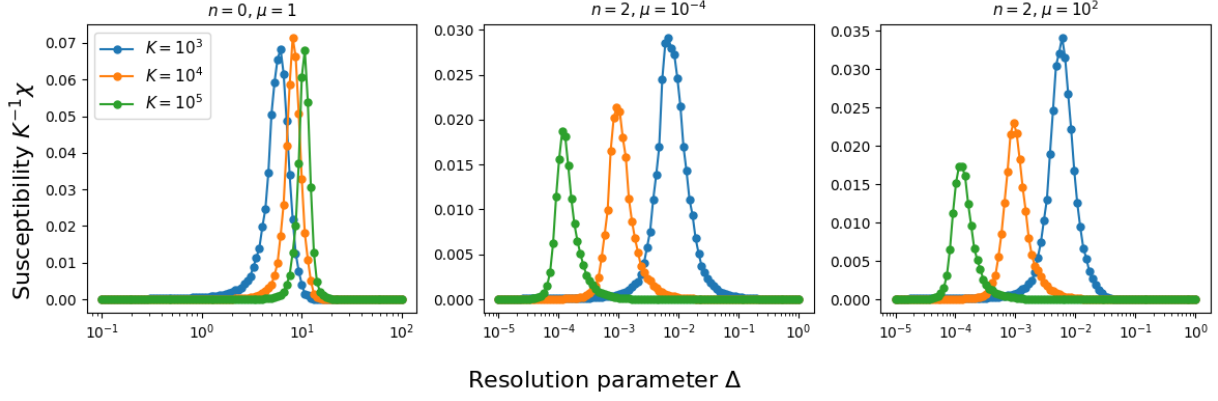


Figure 4.7. Susceptibility χ normalized to K , for different number of events and taking average values for $R = 1000$ realizations.

As we can recognize from both figures, now we have a single transition for $\mu = 10^{-4}$ and $\mu = 10^2$, in principle corresponding to 1D percolation, ergo, the exponents for the size and duration should be $\alpha = \tau = 2$. As we did for the case with $n = 1$, we have studied the avalanches for $K = 10^5$ events and $R = 1000$ realizations to obtain the histograms. The statistics of the avalanches are shown in Figure 4.8.

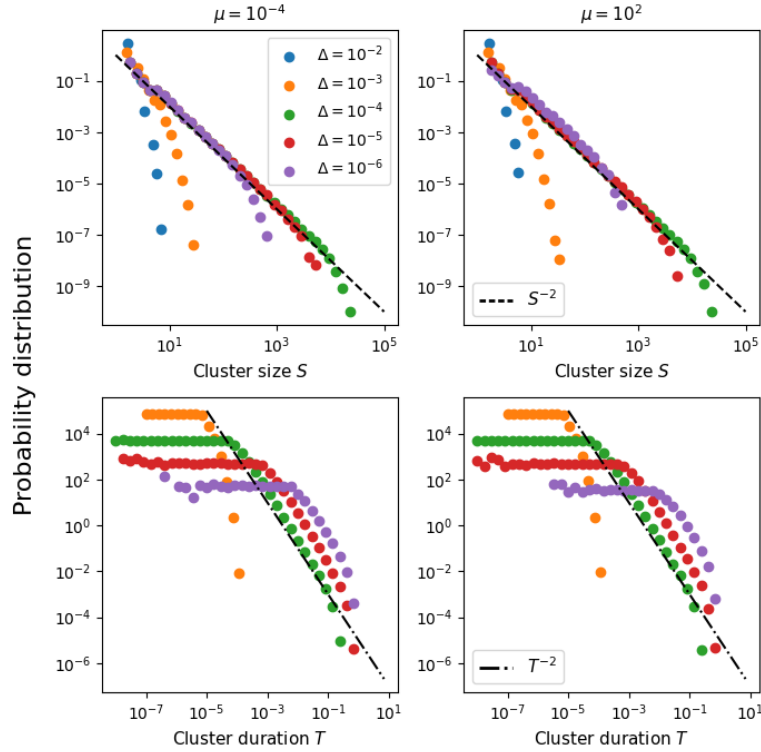


Figure 4.8. Avalanche statistics for a self-exciting Hawkes process with $n = 2$ for $K = 10^5$ events. The histograms have been calculated over $R = 1000$ time series.

As the image shows, we have obtained the exponents $\alpha = \tau = 2$ for both cases, which is compatible with the universality class of 1D percolation. This situation is analogous to the one shown in the case of $n = 1, \mu = 10$ (Figure 1.10) but with a more pronounced effect due to the value of n . Moreover, as happened with $n = 1$, the cutoff of the power-law (caused by the finite size of K) for the cluster duration monotonically shifts to higher values as Δ increases. In conclusion, we observe percolation phenomena for $n \neq 1$, we also observe power law distributions for the size and duration of the avalanches but not caused by criticality as we have seen in the susceptibility diagram, χ only diverges at Δ^* . Note that with $n = 2$, we just have a single transition and its susceptibility peak. For that reason, the criticality is induced only by the order parameter Δ and not by the underlying dynamics, as happened when $n = 1$.

4.3 Results for the bivariate case with excitation and inhibition

We will only look at two parameter configurations. The first one gives us a "pseudo-critical" signal, regulated by inhibition. On the other hand, the second one gives us an oscillatory and "stationary" signal. For both signals, the background rates will be $\mu_E = \mu_I = 10^{-2}$, the other interaction parameters are shown in Table 4.1.

Table 4.1. Interaction parameters for both bivariate processes

	Pseudo-critical	Stationary
n_{EE}	1.5	1.5
n_{EI}	1.5	1.5
n_{IE}	-0.33	-0.5
n_{II}	0	0

With the value of n_{EE} , the signals would be supercritical, but adding the inhibition, the behaviour becomes totally different. Moreover, we will see that a slight change in the inhibition strength will change the signal from pseudo-critical to oscillatory. In Figures 4.9 and 4.10 we present the time series and raster plots for the two different signals.

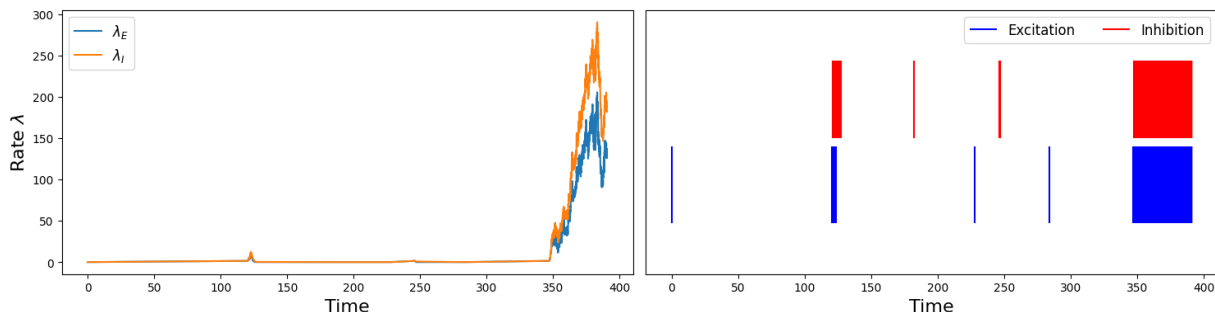


Figure 4.9. On the left, a temporal series of $K = 10^4$ events of a bivariate Hawkes process with the interaction parameters of the pseudo-critical signal shown in Table 4.1, on the right, the raster plot of the same process for excitatory and inhibitory events.

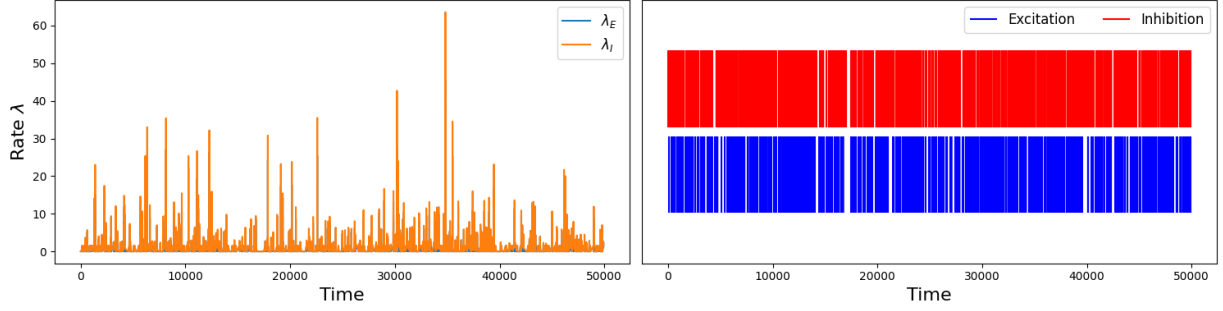


Figure 4.10. On the left, a temporal series of $K = 10^4$ events of a bivariate Hawkes process with the interaction parameters of the “stationary” signal shown in Table 4.1, on the right, the raster plot of the same process for excitatory and inhibitory events.

As illustrated, the pseudo-critical signal has a bursty structure until a big avalanche of activity occurs; on the contrary, the “stationary” signal has a more regular structure.

In this case, we will only cover the case of $\mu \ll 1$. This is because as we have seen in Eq. 3.8 g_j must be positive and having $\alpha_{IE} \equiv n_{IE} < 0$ leads to a negative g_j sporadically. We have solved this issue by setting $\lambda_j(t_k) = \mu_j$ for $j = E, I$, which does not change the dynamics as it does if $\mu_j \gg 1$. In order to cover the case of $\mu \gg 1$ we could use Ogata thinning algorithm [26] to simulate coupled processes, but this algorithm less efficient than the one that we have used and computing the same amount of time series would take much longer.

Now, the main results for the case of excitatory and inhibitory processes are presented. First of all we will show the results for “pseudo-critical” signals shown in Figure 4.9. Both the phase diagrams and avalanche statistics shall be calculated with the event times in general, without distinguishing between excitatory and inhibitory events. Future work could be conducted to study the avalanches for each type of event separately. The phase diagram and its corresponding susceptibility are shown in Figures 4.11 and 4.12 respectively.

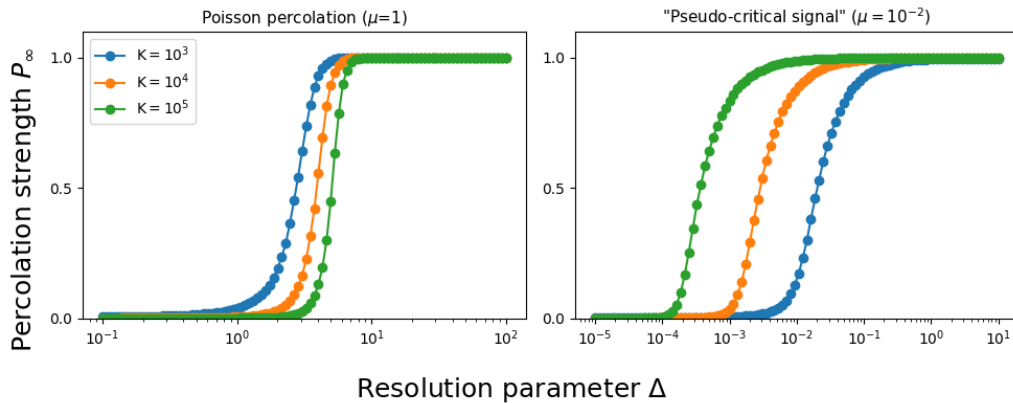


Figure 4.11. Percolation phase diagrams averaged over $R = 1000$ “pseudo-critical” signals of $K = 10^5$ events. ($n_{EE} = n_{EI} = 1.5, n_{IE} = -0.33$)

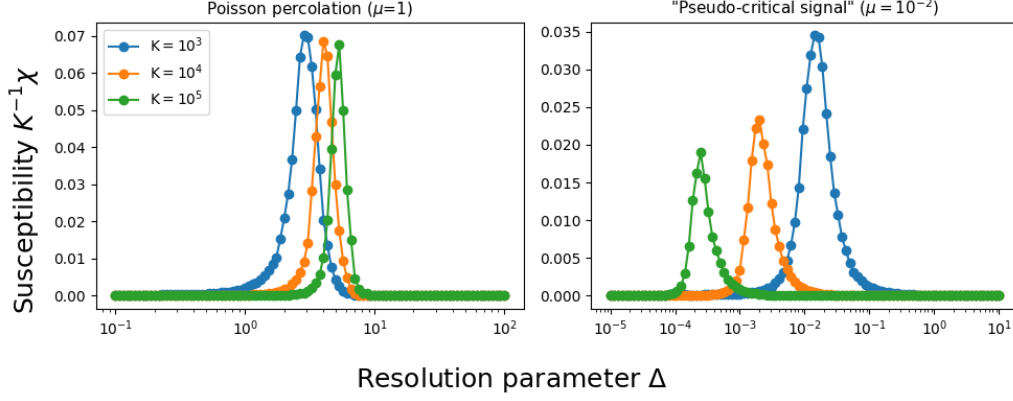


Figure 4.12. Susceptibility χ normalized to the number of events associated with the above phase diagram.

Both figures show a single transition around $\Delta^* \approx 2 \cdot 10^{-4}$ for 10^5 events, that so far has corresponded with the universality class of 1D percolation, but on this occasion, we have the same situation as in the case of $n = 1$ and $\mu = 10^{-4}$, as illustrated in Figure 4.13.

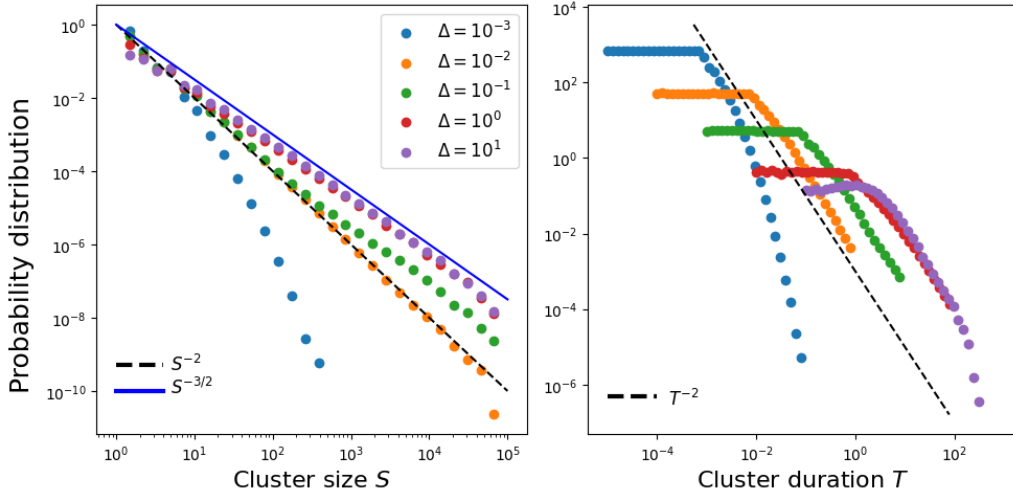


Figure 4.13. Avalanche statistics of $K = 10^5$ events for “pseudo-critical” signals of two coupled Hawkes processes. Histograms have been calculated over $R = 1000$ time series.

In this case, the exponents for the size and duration of the avalanches behave as if there was a double transition in the phase diagram, but in fact, there is not. Further analyses will be needed to clarify this phenomenon. On the contrary, in the case of the “stationary” signal shown in Figure 4.10 we do not have this anomaly. The phase diagram and susceptibility are shown in Figures 4.14 and 4.15, respectively.

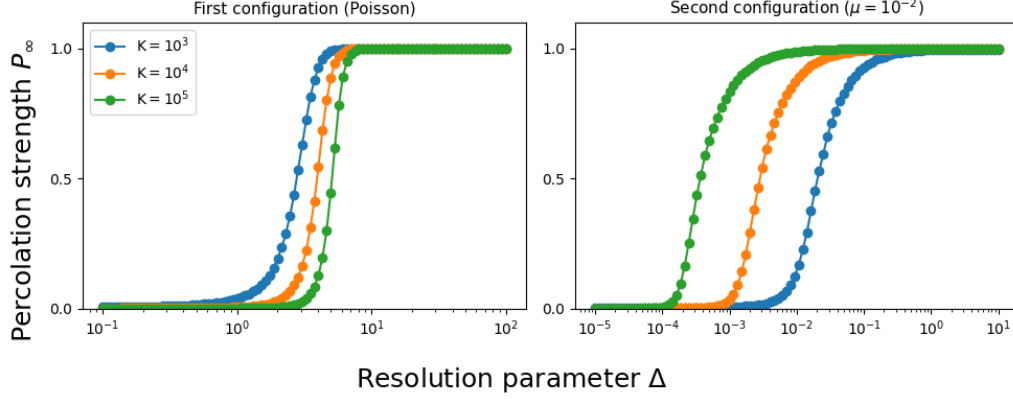


Figure 4.14. Percolation phase diagrams averaged over $R = 1000$ “stationary” signals of $K = 10^5$ events. ($n_{EE} = n_{EI} = 1.5, n_{IE} = -0.5$)

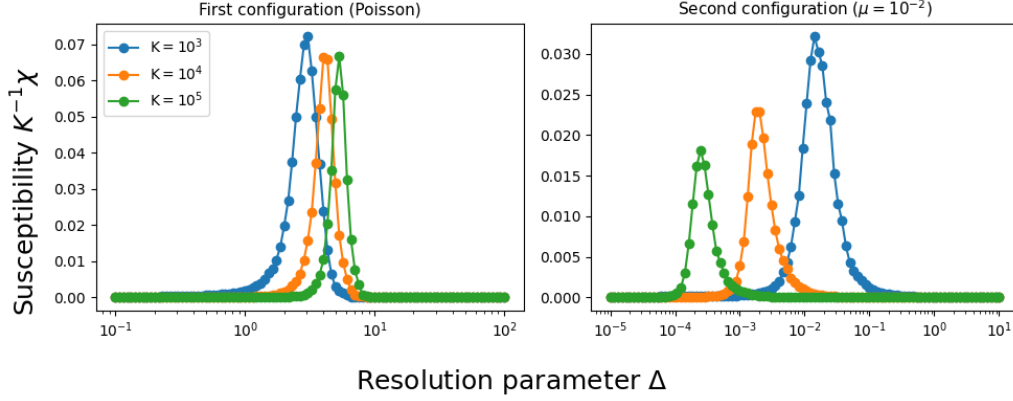


Figure 4.15. Susceptibility χ normalized to K associated with the above phase diagram.

We note an almost identical behaviour to the one shown in the case of the “pseudo-critical” signals, with just a single transition around $\Delta^* \approx 2 \cdot 10^{-4}$ for 10^5 events, agreeing with the universality class of 1D percolation. In this case, as Figure 4.16 indicates, the exponents are indeed $\alpha = \tau = 2$ but exponentially decaying.

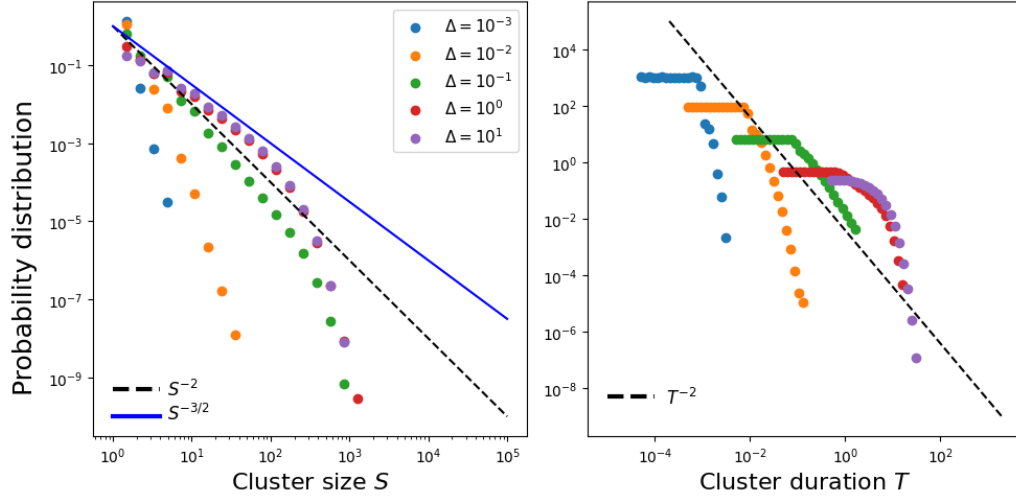


Figure 4.16. Avalanche statistics of $K = 10^5$ events for “stationary” signals of two coupled Hawkes processes. Histograms have been calculated over $R = 1000$ time series.

To conclude the chapter, in Table 4.2 are presented all the exponents obtained for the different configurations studied.

Table 4.2. Power-law exponents for every configuration studied.

	Poisson	n=1		n=2		Bivariate with excitation and inhibiotn	
	$\mu = 1$	$\mu = 10^{-4}$	$\mu = 10^2$	$\mu = 10^{-4}$	$\mu = 10^2$	“Pseudo-critical”	“Stationary”
α	2	$2 \xrightarrow{\Delta \uparrow} 3/2$	2	2	2	$2 \xrightarrow{\Delta \uparrow} 3/2$	~ 2 (damped)
τ	2	2	2	2	2	2	~ 2 (damped)

Chapter 5

Conclusions

To conclude, this master's thesis has brought us closer to the comprehension of criticality emerging from Hawkes processes. We have fulfilled the objectives set at the beginning of the project, starting with a brief overview of criticality in complex systems, with examples in different fields, such as physics or living systems. After a short introduction to point processes, we understood Hawkes processes themselves, how they can be used to model self-exciting events, and how they can be simulated. We also have seen how criticality arises from these processes as long as suitable parameters and time binning are chosen. We have developed computational tools to reproduce the results of the paper by [11] satisfactorily. Moreover, these tools have allowed us to extend the analysis to other configurations, such as the supercritical regime, and realising that the process ceases to be critical.

Furthermore, we also have studied other configurations other than the one presented in the paper, obtaining valuable insights into the system dynamics, specifically, for the case of $n = 2$ there is not intrinsic criticality in the system, the one observed is due to the time binning method. Finally, we have extended the model to account for both excitatory and inhibitory dynamics, and we have repeated the analysis based on the time-binning method to seek for any fingerprints of criticality and to approach a more realistic model of the brain. Further research could be the analysis of the parameter space of two coupled Hawkes processes, the examination of the statistics of excitation and inhibition separately, the study of more processes or the influence of a network structure in the dynamics of the system.

Bibliography

- [1] Miguel A Muñoz. “Colloquium: Criticality and dynamical scaling in living systems”. In: *Reviews of Modern Physics* 90.3 (2018), p. 031001.
- [2] Marco Baiesi and Maya Paczuski. “Scale-free networks of earthquakes and aftershocks”. In: *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 69.6 (2004), p. 066106.
- [3] Romualdo Pastor-Satorras et al. “Epidemic processes in complex networks”. In: *Reviews of modern physics* 87.3 (2015), pp. 925–979.
- [4] Claudio Castellano, Santo Fortunato, and Vittorio Loreto. “Statistical physics of social dynamics”. In: *Reviews of modern physics* 81.2 (2009), p. 591.
- [5] Sofía Aparicio, Javier Villazón-Terrazas, and Gonzalo Álvarez. “A model for scale-free networks: application to twitter”. In: *Entropy* 17.8 (2015), pp. 5848–5867.
- [6] Albert-László Barabási and Réka Albert. “Emergence of scaling in random networks”. In: *science* 286.5439 (1999), pp. 509–512.
- [7] E. Ising. “Contribution to the theory of ferromagnetism”. In: *Z. Phys* 31.1 (1925), pp. 253–258.
- [8] Jens Wilting and Viola Priesemann. “25 years of criticality in neuroscience—established results, open controversies, novel concepts”. In: *Current opinion in neurobiology* 58 (2019), pp. 105–111.
- [9] Theodore Edward Harris et al. *The theory of branching processes*. Vol. 6. Springer Berlin, 1963.
- [10] Dietrich Stauffer and Ammon Aharony. *Introduction to percolation theory*. Taylor & Francis, 2018.
- [11] Daniele Notarmuzi et al. “Percolation theory of self-exciting temporal processes”. In: *Physical Review E* 103.2 (2021), p. L020302.
- [12] Gerardo Chowell et al. “Mathematical models to characterize early epidemic growth: A review”. In: *Physics of life reviews* 18 (2016), pp. 66–97.
- [13] Sandro Azaele et al. “Statistical mechanics of ecological systems: Neutral theory and beyond”. In: *Reviews of Modern Physics* 88.3 (2016), p. 035003.
- [14] Alan J McKane. *Stochastic Processes*. 2009.
- [15] Kiyoshi Kanazawa and Didier Sornette. “Ubiquitous power law scaling in nonlinear self-excited Hawkes processes”. In: *Physical review letters* 127.18 (2021), p. 188301.
- [16] Angelos Dassios and Hongbiao Zhao. “Exact simulation of Hawkes process with exponentially decaying intensity”. In: (2013).
- [17] Patrick J Laub, Young Lee, and Thomas Taimre. *The elements of Hawkes processes*. Springer, 2021.
- [18] Felipe Yaroslav Kalle Kossio et al. “Growing critical: self-organized criticality in a developing neural system”. In: *Physical review letters* 121.5 (2018), p. 058301.
- [19] Wes McKinney. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. “” O’Reilly Media, Inc.”, 2012.

- [20] Jake VanderPlas. *Python data science handbook: Essential tools for working with data.* ” O’Reilly Media, Inc.”, 2016.
- [21] Travis Oliphant et al. *NumPy*. <https://numpy.org/>. Accessed: 2024-06-21. 2024.
- [22] John D. Hunter et al. *Matplotlib*. <https://matplotlib.org/>. Accessed: 2024-06-21. 2024.
- [23] Adrian Barbu, Song-Chun Zhu, et al. *Monte Carlo Methods*. Vol. 35. Springer, 2020.
- [24] Raúl Toral and Pere Colet. *Stochastic numerical methods: an introduction for students and scientists*. John Wiley & Sons, 2014.
- [25] D Stauffer and C Jayaprakash. “Critical exponents for one-dimensional percolation clusters”. In: *Physics Letters A* 64.5 (1978), pp. 433–434.
- [26] Yoshihiko Ogata. “On Lewis’ simulation method for point processes”. In: *IEEE transactions on information theory* 27.1 (1981), pp. 23–31.
- [27] Antonio Rivas Blanco. *TFM*. 2024. URL: <https://github.com/RivasAntonio/TFM>.

Appendix A

Hawkes processes with exponential kernel

As we have seen, the exponential kernel function can be written as: $\phi(t) = \sum_{t_i < t} \alpha e^{-\beta(t-t_i)}$ so the rate becomes:

$$\begin{aligned}\lambda(t) &= \mu + n \sum_{t_i < t} \alpha e^{-\beta(t-t_i)} \\ &= \mu + n \sum_{\substack{t_i < t_k \\ t_k: \text{last event before } t}} \alpha e^{-\beta(t-t_k+t_k-t_i)} \\ &= \mu + n e^{-\beta(t-t_k)} \sum_{t_i < t_k} \alpha e^{-\beta(t_k-t_i)} \\ &= \mu + n e^{-\beta(t-t_k)} (\lambda(t_k) - \mu + \alpha).\end{aligned}\tag{A.1}$$

Where we have used the following expression for the rate of the Hawkes process at time t_k :

$$\lambda(t_k) = \mu + n \sum_{t_i < t_k} \alpha e^{-\beta(t_k-t_i)} \Rightarrow n \sum_{t_i < t_k} \alpha e^{-\beta(t_k-t_i)} = \lambda(t_k) - \mu + \alpha\tag{A.2}$$

Appendix B

Python scripts

Here are the main functions used in the project. The first one is the algorithm used to simulate the inter-event time of a Hawkes process with exponential kernel. The second one is the function for the generation of time series of these Hawkes processes. The third one is the function associated to the calculation of the percolation strength, P_∞ , for the phase diagrams. The fourth one is the function to identify the clusters size and duration distribution. The fifth one is the function to generate the inter-event time of a bivariate Hawkes process with exponential kernel. The last one is the function to generate the time series of the bivariate Hawkes process. Data analysis and plotting functions are not included here, but they can be found in the GitHub repository of the project [27].

Script B.1. Script with the main functions.

```
1 import numpy as np
2
3 def algorithm(rate, mu, n):
4     """
5     Algorithm that computes interevent times and Hawkes intensity for a self-exciting
6     process
7
8     ## Inputs:
9     rate: Previous rate
10    mu: Background intensity
11    n: Weight of the Hawkes process
12
13    ## Outputs: rate x_k, x_k
14    x_k: Inter-event time
15    rate_tk: Intensity at time tk
16    """
17    # 1st step
18    u1 = np.random.uniform()
19    if mu == 0:
20        F1 = np.inf
21    else:
22        F1 = -np.log(u1) / mu
23
24    # 2nd step
25    u2 = np.random.uniform()
26    if (rate - mu) == 0:
27        G2 = 0
28    else:
29        G2 = 1 + np.log(u2) / (rate - mu)
30
31    # 3rd step
```

```

31     if G2 <= 0:
32         F2 = np.inf
33     else:
34         F2 = -np.log(G2)
35
36     # 4th step
37     xk = min(F1, F2)
38
39     # 5th step
40     rate_tk = (rate - mu) * np.exp(-xk) + n + mu
41     return rate_tk, xk
42
43 def generate_series(K, n, mu):
44     """
45     Generates temporal series for K Hawkes processes
46
47     ## Inputs:
48     K: Number of events
49     n: Strength of the Hawkes process
50     mu: Background intensity
51
52     ## Outputs:
53     times_between_events: time series the inter-event times
54     times: time series of the events
55     rate: time series for the intensity
56     """
57     times_between_events = [0]
58     rate = [mu]
59     for _ in range(K):
60         rate_tk, xk = algorithm(rate[-1], mu, n)
61         rate.append(rate_tk)
62         times_between_events.append(xk)
63     times = np.cumsum(times_between_events)
64     return times_between_events, times, rate
65
66 def calculate_percolation_strength(times_between_events, deltas):
67     """
68     Calculate the percolation strength for a given set of deltas (resolution parameters)
69
70     ## Inputs:
71     times_between_events: time series of interevent times
72     deltas: list of resolution parameters
73
74     ## Output:
75     percolation_strengths: list of percolation strengths
76     """
77     percolation_strengths = []
78
79     for delta in deltas:
80         cluster_sizes = []
81         current_cluster_size = 1 # The first event is always a cluster
82
83         for time in times_between_events:
84             if time < delta:
85                 current_cluster_size += 1
86             else:
87                 if current_cluster_size > 1: # Only consider clusters with more than one
event
88                     cluster_sizes.append(current_cluster_size)
89                 current_cluster_size = 1 # The next event is always a cluster
90

```

```

91         if current_cluster_size > 1: # Consider the last cluster if it ends at the last
event
92             cluster_sizes.append(current_cluster_size)
93
94         if len(cluster_sizes) != 0: # Check if cluster_sizes is not empty to avoid
errors
95             max_cluster_size = max(cluster_sizes)
96         else:
97             max_cluster_size = 0
98
99         percolation_strengths.append(max_cluster_size / len(times_between_events))
100
101     return percolation_strengths
102
103
104 def identify_clusters(times, delta):
105     """
106     Identifies clusters in a temporal series given a resolution parameter delta
107     Computes the size and duration of clusters
108
109     ## Inputs:
110     times: temporal series
111     delta: resolution parameter
112
113     ## Outputs:
114     clusters: list of clusters
115     clusters_sizes: list of sizes of clusters
116     clusters_times: list of clusters durations
117     """
118     clusters = []
119     current_cluster = []
120     for i in range(len(times) - 1):
121         if times[i + 1] - times[i] <= delta:
122             if not current_cluster:
123                 current_cluster.append(times[i])
124                 current_cluster.append(times[i + 1])
125             else:
126                 if current_cluster:
127                     clusters.append(current_cluster)
128                     current_cluster = []
129
130     clusters_sizes = [len(cluster) for cluster in clusters]
131     clusters_times = [cluster[-1] - cluster[0] for cluster in clusters]
132     return clusters, clusters_sizes, clusters_times
133
134 def bivariate_algorithm(rate1, rate2, muE, muI, nEE, nII, nEI, nIE):
135     """
136     Algorithm that computes interevent times and Hawkes intensity for a bivariate Hawkes
process
137
138     ## Inputs:
139     rate1: Previous excitation rate
140     rate2: Previous inhibition rate
141     nEE: "Strength" of the autoexcitation process
142     nII: "Strength" of the autoinhibition process
143     nEI: "Strength" of the excitation to the inhibition
144     nIE: "Strength" of the inhibition to the excitation
145     muE: Background intensity of the excitation
146     muI: Background intensity of the inhibition
147
148
149     ## Outputs:

```

```

150     rate1_tk: Intensity of the excitation at time tk
151     rate2_tk: Intensity of the inhibition at time tk
152     x_k: Inter-event time
153     reaction: (0 for excitatory events and 1 for inhibitory events)
154     """
155     _, xk1 = algorithm(rate1, muE, nEE)
156     _, xk2 = algorithm(rate2, muI, nII)
157
158     xks = [xk1, xk2]
159
160     reaction = np.argmin(xks)
161
162     rate1_tk = 0.
163     rate2_tk = 0.
164
165     if reaction == 0:
166         rate1_tk = (rate1 - muE) * np.exp(-xk1) + nEE + muE
167         rate2_tk = (rate2 - muI) * np.exp(-xk1) + nEI + muI
168     else:
169         rate1_tk = (rate1 - muE) * np.exp(-xk2) + nIE + muE
170         rate2_tk = (rate2 - muI) * np.exp(-xk2) + nII + muI
171
172
173     if rate1_tk <= muE:
174         rate1_tk = muE
175     if rate2_tk <= muI:
176         rate2_tk = muI
177
178     xk = xks[reaction]
179
180     return rate1_tk, rate2_tk, xk, reaction
181
182 def generate_series_bivariate(K, nEE, nII, nEI, nIE, muE, muI):
183     """
184     Generates temporal series for K bivariate Hawkes processes
185
186     ## Inputs:
187     K: Number of events
188     nEE: "Strength" of the autoexcitation process
189     nII: "Strength" of the autoinhibition process
190     nEI: "Strength" of the excitation to the inhibition
191     nIE: "Strength" of the inhibition to the excitation
192     muE: Background intensity of the excitation
193     muI: Background intensity of the inhibition
194
195     ## Outputs:
196     times_between_events: time series the interevent times
197     times: time series the events
198     rate1: time series for the intensity of process 1 (Excitation)
199     rate2: time series for the intensity of process 2 (Inhibition)
200     reactions: list the event type (0 for excitation. 1 for inhibition)
201     """
202     times_between_events = [0]
203     rate1 = [muE]
204     rate2 = [muI]
205     reactions = []
206     for _ in range(K):
207         rate1_tk, rate2_tk, xk, reaction = bivariate_algorithm(rate1[-1], rate2[-1], muE,
208             muI, nEE, nII, nEI, nIE)
209         rate1.append(rate1_tk)
210         rate2.append(rate2_tk)
211         reactions.append(reaction)

```

```
211         times_between_events.append(xk)
212     times = np.cumsum(times_between_events)
213
214     return times_between_events, times, rate1, rate2, reactions
```