EPFL

Thèse n° 7143

Learning Self-Exciting Temporal Point Processes Under Noisy Observations

Présentée le 1er octobre 2021

Faculté informatique et communications
Laboratoire de la Dynamique de l'Information et des Réseaux 1
Programme doctoral en informatique et communications

pour l'obtention du grade de Docteur ès Sciences

par

William TROULEAU

Acceptée sur proposition du jury

Prof. M. Salathé, président du jury
Prof. M. Grossglauser, Prof. P. Thiran, directeurs de thèse
Prof. N. He, rapporteuse
Prof. M. Gomez Rodriguez, rapporteur
Prof. D. Kuhn, rapporteur

École polytechnique fédérale de Lausanne

2021

The key question to keep asking is, Are you spending your time on the right things? Because time is all you have.

— Randy Pausch

Acknowledgements

The amazing journey that was my research for this thesis would not have been possible without the constant support of many people! Before expressing my gratitude to the people named individually hereafter, I would like to thank all those who are not mentioned, including the friends I made along the way, my numerous colleagues at EPFL, and the students I advised and assisted.

First and foremost, I would like to thank my advisors, Prof. Matthias Grossglauser and Prof. Patrick Thiran for their continuous support throughout the years. You always had my back at all times and you gave me, without restraints, the opportunity to explore research directions close to my heart. Your consistent faith in me pushed me to overcome all the obstacles and enabled me to grow as a researcher. With the combination of Patrick's precision and Matthias's creativity, our weekly meetings were an endless source of inspiration. I am also grateful for the opportunity to have assisted you in teaching. It was never easy, but it was an incredibly rewarding experience that I will surely miss. You are both truly amazing professors.

I would like to thank the members of my jury committee, Niao He, Manuel Gomez-Rodriguez, Daniel Kuhn, and Marcel Salathé. Thank you for your time, your feedback, and the constructive discussions we had on my defense.

This thesis would not have been possible without Negar Kiyavash and Jalal Etesami. Meeting you during your visit at EPFL in 2017 was a turning point in my research, and I will be forever thankful for the opportunity to have worked with you. Your constant cheerful spirits and stimulating discussions pushed my research to another level.

I am especially grateful for the fantastic staff of our lab who made everything run smoothly for me. Thank you, Patricia and Angela, for your smiles and enthusiasm that always illuminated my days. Thank you, Holly, for your "hollifying" editing magic. When reviewers said "the paper was a pleasure to read", the credit comes down to you. I also want to thank Sylviane who, since day one, helped me many times. I feel lucky to have had the opportunity to help you in assisting new students for a few years.

I extend a thank you to all my lab mates and friends in the INDY lab. Victor, our "coffain" sessions and beers at Sat were single-handed in keeping me afloat. Christina,

Lucas, and Jalal, thank you for being the best office mates I could wish for. Mohamed and Vincent, thank you for your mentorship that motivated me to take this path. Thank you Farnood, Daniyar, Arnout, Mahsa, Alexandre, Mladen, Greg, Aswin, Brunella, Young-Jun, Ehsan, Sébastien. I will miss our long coffee-breaks, lunches, and ski outings.

Many thanks go to all my friends for reminding me that there is a life worth living outside of academia. *Merci vielmal* for all the fun times, Lisa, Marzell, Pierrot, Laurie, Ilia, Ben and all my EPFL friends. A special thank you goes to Pierre and Louis for always being there for me. Thank you, Salman, for being an inspiration for now more than a decade. The long discussions I had with you and your father inspired me to take on the research path. Thank you, the Earle family. David, Henri, Duncan and Robin, you have welcomed me into your home countless times since my childhood. You are the role models who guided me in the choice of my research directions. Thank you,Vince and Sandhya. I lived with you for less than a year, but your positive influence will remain forever. Thank you also go to the incredible Etter family. Vincent, Sophie, Sarah, and Marie, you made my time in Seattle so enjoyable.

Most importantly, I would like to thank my family. Thank you, *Papa et Maman*. While always having my back, you let me choose my own path and gave me the opportunity to study for all these years and to follow my dreams. Thank you, the Gouiller family, for welcoming me as your newest member. Last but not least, thank you, Justine, for the consistent support you brought me every step of the way. Thank you for bringing love and happiness to every day. I would never have made it so far without you. You make me want to be a better person every day. I know that, with you by my side, everything becomes possible.

Lausanne, June 24, 2021 William Trouleau

Abstract

Understanding the diffusion patterns of sequences of interdependent events is a central question for a variety of disciplines. *Temporal point processes* are a class of elegant and powerful models of such sequences; these processes have become popular across multiple fields of research due to the increasing availability of data that captures the occurrence of events over time. A notable example is the Hawkes process. It was originally introduced by Alan Hawkes in 1971 to model the diffusion of earthquakes and was subsequently applied across fields such as epidemiology, neuroscience, criminology, finance, genomic, and social-network analysis.

A central question in these fields is the inverse problem of uncovering the diffusion patterns of the events from the observed data. The methods for solving this inverse problem assume that, in general, the data is noiseless. However, real-world observations are frequently tainted by noise in a number of ways. Most existing methods are not robust against noise and, in the presence of even a small amount of noise in the data, they might completely fail to recover the underlying dynamics. In this thesis, we remedy this shortcoming and address this problem for several types of observational noise.

First, we study the effects of small event-streams that are known to make the learning task challenging by amplifying the risk of overfitting. Using recent advances in variational inference, we introduce a new algorithm that leads to better regularization schemes and provides a measure of uncertainty on the estimated parameters.

Second, we consider events corrupted by unknown synchronized time delays. We show that the so-called *synchronization noise* introduces a bias in the existing estimation methods, which must be handled with care. We provide an algorithm to robustly learn the diffusion dynamics of the underlying process under this class of synchronized delays.

Third, we introduce a wider class of random and unknown time shifts, referred to as *random translations*, of which synchronization noise is a special case. We derive the statistical properties of Hawkes processes subject to random translations. In particular, we prove that the cumulants of Hawkes processes are invariant to random translations and we show that cumulant-based algorithms can be used to learn their underlying causal structure even when unknown time shifts distort the observations.

Abstract

Finally, we consider another class of temporal point processes, the so-called *Wold* process that solves a computational limitation of the Bayesian treatment of Hawkes processes while retaining similar properties. We address the problem of learning the parameters of a Wold process by relaxing some of the restrictive assumptions made in the state of the art and by introducing a Bayesian approach for inferring its parameters.

In summary, the results presented in this dissertation highlight the shortcomings of standard inference methods used to fit temporal point processes. Consequently, these results deepen our ability to extract reliable insights from networks of interdependent event streams.

Keywords event streams, noisy observations, temporal point processes, Hawkes process, Granger causality, networks, algorithms, Bayesian modeling, statistical inference, machine learning

Résumé

Comprendre les schémas de diffusion de séquences d'événements interdépendants est une question centrale pour une variété de disciplines scientifiques. Les processus ponctuels constituent une classe de modèles élégants et puissants pour de telles séquences temporelles. Ils sont devenus populaires dans de nombreux domaines de recherche en raison de la disponibilité croissante de données qui capturent l'occurrence d'événements dans le temps. Un exemple notable est le processus de Hawkes qui a été introduit par Alan Hawkes en 1971 pour modéliser la diffusion des tremblements de terre et a ensuite été appliqué à de nombreux domaines tels que l'épidémiologie, les neurosciences, la criminologie, la finance, la génomique et l'analyse des réseaux sociaux.

Une question centrale dans ces domaines est le problème inverse qui consiste à découvrir les structures du système de diffusion des événements à partir d'observations passées. Les méthodes s'attaquant à ce problème inverse supposent en général que les évènements soient observés sans bruit. Cependant, dans des applications réelles, les données observées sont fréquemment entachées de plusieurs sources d'erreurs. La plupart des méthodes existantes ne sont pas robustes contre ces erreurs et, même en présence d'une quantité minime de bruit dans les données, elles peuvent complètement échouer à inférer les schémas de diffusion des séquences d'événements. Dans cette thèse, nous remédions à cette lacune et abordons ce problème pour plusieurs types de bruit.

Tout d'abord, nous étudions l'effet de petits flux d'évènements qui sont connus pour rendre la tâche d'apprentissage difficile en amplifiant le risque de surapprentissage (aussi appelé "*overfitting*"). En utilisant les progrès récents en inférence variationnelle, nous introduisons un nouvel algorithme qui conduit à de meilleures méthodes de régularisation et fournit une mesure de l'incertitude sur les paramètres estimés.

Deuxièmement, nous considérons des événements corrompus par des décalages temporels synchronisés et inconnus. Nous montrons que ce bruit de synchronisation introduit un biais dans les méthodes d'estimation existantes et doit donc être traité avec précaution. Nous proposons un algorithme pour apprendre de manière robuste les dynamiques de diffusion du processus sujets à cette classe de retards synchronisés.

Troisièmement, nous introduisons une classe plus large de décalages temporels aléatoires et inconnus, appelée translations aléatoires, dont le bruit de synchronisation est un cas particulier. Nous dérivons les propriétés statistiques des processus de Hawkes soumis à des translations aléatoires. En particulier, nous prouvons que les cumulants des processus de Hawkes demeurent invariants aux translations aléatoires et nous montrons que les algorithmes basés sur les cumulants peuvent être utilisés pour apprendre leur structure causale sous-jacente même lorsque des décalages temporels inconnus déforment les observations.

Enfin, nous considérons une autre classe de processus ponctuels, les processus dits de Wold, qui résolvent une limitation computationnelle du traitement Bayésien des processus de Hawkes tout en conservant des propriétés similaires. Nous abordons le problème de l'apprentissage des paramètres d'un processus de Wold en relaxant certaines des hypothèses restrictives faites dans les modèles existants et en introduisant une approche Bayésienne pour inférer ses paramètres.

En résumé, les résultats présentés dans cette thèse mettent en évidence les lacunes des méthodes d'inférence standard utilisées pour ajuster les processus temporels ponctuels. Par conséquent, ces résultats renforcent notre capacité à extraire des informations fiables de réseaux de flux d'événements interdépendants.

Mots-clés flux d'évènements, processus ponctuels et temporels, processus de Hawkes, causalité de Granger, réseaux, algorithmes, modèles Bayésiens, inférence statistique, apprentissage automatique

Contents

Mathematical Notation

Symbol	Description
x, X	Plain letters denote scalar values.
$\boldsymbol{x} = [x_i]$	Boldface lowercase letters denote column vectors.
$\boldsymbol{X} = [x_{ij}]$	Boldface uppercase letters denote matrices.
$\mathcal{X}$	Calligraphic uppercase letters denote sets.
$\mathbb{R}$, $\mathbb{R}_{>0}$, $\mathbb{N}$, $\mathbb{N}_0$	Number types: real, positive real, natural numbers, natural numbers starting from 0, respectively.
$[n]$	Set of consecutive natural numbers $\{1, \ldots, n\}$.
$\mathbb{P}(\mathcal{A})$	Probability of the random event $\mathcal{A}$.
$\mathbb{1}_{\{\mathcal{A}\}}$	Indicator variable of the random event $\mathcal{A}$.
$\mathbb{E}[x]$	Expectation of the random variable x.
$f * g(t)$	Convolution between two scalar functions, defined as $f * g(t) \triangleq \int_{\mathbb{R}} f(t-u)g(u)du$.
$\mathcal{L}[f](\boldsymbol{s})$	Laplace transform of a function $f(\boldsymbol{x})$, defined as $\mathcal{L}[f](\boldsymbol{s}) \triangleq \int_{\mathbb{R}^n} f(\boldsymbol{x}) \exp(-\boldsymbol{s}^T\boldsymbol{x})d\boldsymbol{x}$.
$O(f(x))$	$g(x) = O(f(x)) \iff \limsup_{x\to\infty} \lvert g(x)\rvert / f(x) < \infty$.
$o(f(x))$	$g(x) = o(f(x)) \iff \lim_{x\to\infty} g(x)/f(x) = 0$.
$\Omega(f(x))$	$g(x) = \Omega(f(x)) \iff f(x) = O(g(x))$.
$\omega(f(x))$	$g(x) = \omega(f(x)) \iff f(x) = o(g(x))$.

Continuous Distribution	Domain	Probability Density function $f(x)$
Exponential(λ)	$\mathbb{R}_{>0}$	$\lambda \exp(-\lambda x)$
Gamma(α, β)	$\mathbb{R}_{>0}$	$\frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \exp(-\beta x)$
Inverse$-$Gamma(α, β)	$\mathbb{R}_{>0}$	$\frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \exp(-\frac{\beta}{x})$
Uniform(a, b)	$[a, b]$	$\frac{1}{b-a}$
Normal(μ, σ)	$\mathbb{R}$	$\frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$
Lognormal(μ, σ)	$\mathbb{R}_{>0}$	$\frac{1}{\sqrt{2\pi}\sigma x} \exp\left[-\frac{(\log x-\mu)^2}{2\sigma^2}\right]$

Discrete Distribution	Domain	Probability Mass function $\mathbb{P}(X = k)$
Poisson(λ)	$\mathbb{N}_0$	$\frac{\lambda^k}{k!} \exp(-\lambda)$
Categorical($p_1, \ldots, p_n$)	$[n]$	p_k

1 Introduction

1.1 Motivation

Analyzing the precise time interval between the occurrences of a natural or social phenomenon has been a central question to scientists for centuries. As early as in the 1600s in the city of London, the Church of England began publishing weekly *bills of mortality* to monitor burials of parishioners. In 1662, John Graunt analyzed 70 years' worth of this data and reported time trends for many diseases in the *Natural and Political Observations on the Bills of Mortality*. This publication was a pioneering work providing statistical evidence for many theories on diseases. It thereby laid the foundations of the field of statistics and granted John Graunt the title of first epidemiologist[1]. The publication consists of the first *life table* (also called mortality table); it compiles the survival of people from a certain population in order to quantitatively measure their longevity.

The emergence of this type of *event data* that records occurrences of a phenomenon over time led to the development of *stochastic point processes*, a mathematical framework describing random sets of discrete points scattered in some space. The basic idea behind point processes is to capture the expected arrival rate of events, called *intensity function*. In recent decades, point processes have received a steady increase of popularity and have become an essential chapter of the theory of stochastic processes. They appear in various forms in many applications, from epidemiology to telecommunication networks, biology, finance or social sciences.

In this thesis, we are interested in the problem of learning the temporal dynamics of *marked temporal point processes*, *i.e.*, point processes tailored for event data consisting of

[1] Morabia [91] provides an interesting history of modern epidemiology and more details on John Graunt's work.

one or more sequences of events of the form

$$\mathcal{S} = \{(i_n, t_n)\}_{n \geq 1},$$

where $t_n \in \mathbb{R}$ is the time of occurrence of the n-th event and $i_n \in \mathcal{X}$ is its type, or mark, which belongs to a discrete set $\mathcal{X}$. The mark might, for example, represent a geographical location, a user in a social network, or an individual in a population.

A typical task of interest is to extract the pairwise influence relationships between types of interdependent events. Over the years, a class of point process emerged as a central model to capture these relationships: self-exciting point processes, also called *Hawkes processes*, which capture the self- and the mutual-excitation patterns between several types of events. Even though self-exciting point processes were originally developed by Hawkes [58] to model the diffusion of earthquakes, they rapidly found applications in other disciplines. In particular, they were applied to model the firing patterns of nerve cells in the brain [95], to study the extremal returns in high-frequency trading [11], to uncover the epidemic pathways of infection diseases [65], and to forecast and control opinion dynamics in social networks [48]. One of the reasons behind the popularity of Hawkes processes is that their dynamics can be elegantly summarized into a directed network that captures the Granger causality between the streams of events, a statistical measure of causality that is based on predictiveness.

Handling Noisy Observations. Learning the dynamics of event data with Hawkes processes is an active research problem. A large body of recent studies exploit advances in deep learning and focus on designing increasingly complex intensity functions for applications where large volumes of event data are available. However, very few efforts discuss the systematic noise that may taint the observed events in a number of ways, as well as the various ways in which this noise affects the learning algorithms of classic models such as the Hawkes process. In this thesis, we remedy this shortcoming and address this problem for several types of observational noises.

- **Small Data.** Although many applications benefit from the growing availability of data from the web, some applications still have to work with small datasets comprised of a limited number of events. Consider, for example, the problem of learning the diffusion pathways of an infectious disease. The goal of policy makers is to take actions as quickly as possible, as a result limiting the amount of observed data. In other public health applications, data is typically collected manually through surveys, which requires expensive field work. It is therefore crucial to develop algorithms that are able to uncover the dynamics of the process when only a handful of events is available.

- **Random Time Shifts.** The process through which the sequences of events are collected often introduces noise in the timestamps of the observed events. For instance, in neuroscience, the activity of neurons is typically collected by measuring a continuous signal coming from the action potential of neurons by using electrode micro-arrays. The signal is then converted into discrete sequences of events of firing neurons, called neuronal spike trains; they are the times when the action potential exceeds a threshold. This procedure is inherently noisy and prone to introducing inaccuracies in the measured timestamps. Another example is in epidemiology, where the reported times of infection have an approximate granularity and do not account for the latent incubation period. This could lead to inaccuracies in the measured timestamps. Consequently, a secondary case might be reported before the primary case, which could interfere with learning the true line of causation. Most existing methods for learning point processes are based on a likelihood function that relies on the order of events to extract the underlying dynamics of the processes, and hence lack robustness to noise that affects timing information, even in small amounts.

- **Scalability without compression.** Some applications such as social network analysis and information diffusion deal with very big networks containing a large number of event types. In such settings, mining the Granger causality graph of Hawkes processes becomes a challenging task. In particular, the Bayesian treatment of the Hawkes process is known to not scale well with both the number events and the number of dimensions [77]. The only solution proposed to overcome this challenge consists of a discretization of binning the events through a discretization of time, at the expense of information loss. It is therefore essential to develop algorithms to capture the Granger causality in large networks of multivariate temporal point processes with introducing noise to the observed data.

In the following chapters of this thesis, we tackle in turn each of the aforementioned challenges, and we characterize their impact on estimation procedures for self-exciting temporal point processes. Before delving into these questions, we first provide an overview of the temporal point process framework used throughout the dissertation.

1.2 An Overview of Temporal Point Processes

Although there are excellent textbooks, such as the classic *An Introduction to the Theory of Point Processes* from Daley and Vere-Jones [34, 35], that discuss point processes in extensive detail, this section is meant to be a comprehensive introduction without delving too deep into technicalities of measure theory. To make it easier to grasp the intuition behind the concepts used in such models, we first ignore the marks of the events and begin by considering only their purely temporal information. We introduce the basic

definitions from the perspective of counting problems. We then illustrate these definitions with a few classic examples.

1.2.1 Preliminary Definitions

Consider a sequence of *events* occurring in a system. For example, an event can be a transaction in a financial market, the transmission of an infectious disease in an epidemic, a post on a social network, or a neuron firing in the brain. A temporal point process is a probabilistic representation of these events and is formally defined as follows.

Definition 1.1 (Temporal Point Process)**.** A sequence $\mathcal{T} = \{t_n\}_{n\geq 1}$ of real, positive and strictly increasing random variables, defined on $[0, T]$, describing the time of occurrence of a certain event in a system, is called a *temporal point process* on $[0, T]$.

By superposing the times of all events, we obtain the cumulative count of events that occurs over time. This representation provides another way to characterize the system by a so-called *counting process*, defined as follows.

Definition 1.2 (Counting Process)**.** Consider a temporal point process $\mathcal{T} = \{t_n\}_{n\geq 1}$. The stochastic process $\{N(t)\}_{t\in\mathbb{R}_{\geq 0}}$ with right-continuous sample paths defined as

$$N(t) \coloneqq \sum_{n\geq 1} \mathbb{1}_{\{t\geq t_n\}}, \tag{1.1}$$

is the *counting process* associated with the temporal point process $\mathcal{T} = \{t_n\}_{n\geq 1}$. Furthermore, we denote by $\mathcal{H}_t$ the history of the process up to time t, which contains all the event times in $\mathcal{T}$ up to time t.

It is easy to see that both definitions are equivalent. Indeed, (1.1) means that the differential of the counting process, defined as

$$dN(t) \coloneqq N(t + dt) - N(t) \tag{1.2}$$

is equal to 1 for all $t \in \mathcal{T}$ and is equal to 0 otherwise. Definition 1.2 also implies that $N(0) = 0$. Specifically, no event has yet occurred by time $t = 0$. This is an arbitrary choice made for simplification. By abuse of notation, both the sequence of event times $\mathcal{T} = \{t_n\}_{n\geq 1}$ and the counting process $\{N(t)\}_{t\in\mathbb{R}_{\geq 0}}$ are often referred to as point processes because they both carry the same information. We show an illustrative example of a counting process in Figure 1.1.

A natural way to define the density of events at a given time t is through the expected rate of arrival of events within the infinitesimal interval $(t, t + dt]$, called the *conditional intensity function* and formally defined as follows.

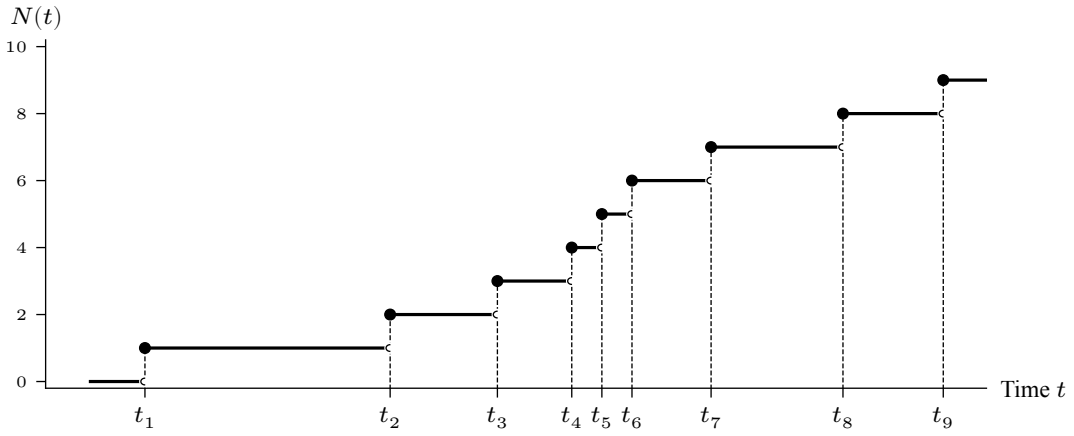


Figure 1.1 – Example of a realization of a point process with events at times $t_1, t_2, \ldots, t_9$ and its corresponding counting process $N(t)$.

Definition 1.3 (Conditional Intensity Function)**.** The conditional intensity function of a temporal point process is given by

$$\lambda(t|\mathcal{H}_t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}\left[N(t+\Delta t) - N(t) \mid \mathcal{H}_t\right]}{\Delta t}. \tag{1.3}$$

In the literature, the conditioning of $\lambda(t|\mathcal{H}_t)$ on the history $\mathcal{H}_t$ is often omitted to simplify the notation. We follow this convention and abbreviate the conditioning with an asterix, writing $\lambda^*(t) \coloneqq \lambda(t|\mathcal{H}_t)$. where applicable.

The conditional probability, given the history $\mathcal{H}_t$, of a new event to occur in an interval $(t, t+dt]$ is then characterized by the conditional intensity function such that

$$\mathbb{P}(dN(t) = 1 \mid \mathcal{H}_t) = \lambda(t \mid \mathcal{H}_t)dt, \tag{1.4}$$

$$\mathbb{P}(dN(t) > 1 \mid \mathcal{H}_t) = o(dt). \tag{1.5}$$

Assumption (1.5) states that the probability of two or more events happening simultaneously is negligible. It is the assumption of so-called *simple* point processes; it holds true for all point processes discussed in this thesis. It follows that, for such processes, the random variable $dN(t)$ is a Bernoulli random variable (up to first order), hence

$$\mathbb{E}[dN(t)] = \mathbb{P}(dN(t) = 1) = \mathbb{P}(\text{“an event occurred at time } t\text{”}). \tag{1.6}$$

Alternatively, it is useful to relate the conditional intensity function $\lambda^*(t)$ to the distribution of time intervals between consecutive events, or *inter-event time* distribution. Let $f(t_{n+1}|\mathcal{H}_{t_n})$ be the density function, and $F(t_{n+1}|\mathcal{H}_{t_n})$ be cumulative distribution, of the

next event t_{n+1} given the history of previous events, formally defined for $t > t_n$ as

$$f(t|\mathcal{H}_{t_n}) = \lim_{dt \to 0} \mathbb{P}\left(t_{n+1} \in (t, t+dt] \mid \mathcal{H}_{t_n}\right)/dt, \tag{1.7a}$$

$$F(t|\mathcal{H}_{t_n}) = \mathbb{P}\left(t_{n+1} \in (t_n, t] \mid \mathcal{H}_{t_n}\right). \tag{1.7b}$$

We can rewrite the conditional intensity function defined in (1.4) in terms of (1.7), as

$$\begin{aligned}
\lambda^*(t)dt &= \mathbb{P}\left(t_{n+1} \in (t, t+dt]|\mathcal{H}_t\right) \\
&= \mathbb{P}\left(t_{n+1} \in (t, t+dt]|\mathcal{H}_{t_n}, t_{n+1} \notin (t_n, t]\right) \\
&= \frac{\mathbb{P}\left(t_{n+1} \in (t, t+dt], t_{n+1} \notin (t_n, t]|\mathcal{H}_{t_n}\right)}{\mathbb{P}\left(t_{n+1} \notin (t_n, t]|\mathcal{H}_{t_n}\right)} \\
&= \frac{\mathbb{P}\left(t_{n+1} \in (t, t+dt]|\mathcal{H}_{t_n}\right)}{\mathbb{P}\left(t_{n+1} \notin (t_n, t]|\mathcal{H}_{t_n}\right)}.
\end{aligned}$$

Hence, the conditional intensity function can be written as

$$\lambda^*(t) = \frac{f(t|\mathcal{H}_{t_n})}{1 - F(t|\mathcal{H}_{t_n})}. \tag{1.8}$$

Overall, the conditional inter-event time distribution and the conditional intensity function are two equivalent ways to uniquely determine the probability structure of a temporal point process. We refer the interested reader to Propositions 7.2.I and 7.2.IV in [34] for the formal proof.

As shown by Daley and Vere-Jones [34, Proposition 7.2.III], the joint density of events of a point process, also called the *likelihood function*, can be expressed in terms of its conditional intensity function, as follows.

Proposition 1.4 (Likelihood Function)**.** *Let N be a point process on $[0, T]$ for some finite positive T, and let $t_1, \ldots, t_{N(T)}$ denote a realization of N over $[0, T]$. Then the likelihood of such N can be written as*

$$\mathcal{L} = \left[\prod_{i=1}^{N(T)} \lambda^*(t_i)\right] \exp\left(-\int_0^T \lambda^*(u)du\right). \tag{1.9}$$

Proof. The expression of the likelihood function can be obtained from the inter-event time distribution in (1.7). As shown by Rasmussen [100], we can rearrange (1.8) into

$$\lambda^*(t) = \frac{f(t|\mathcal{H}_{t_n})}{1 - F(t|\mathcal{H}_{t_n})} = \frac{\frac{d}{dt}F(t|\mathcal{H}_{t_n})}{1 - F(t|\mathcal{H}_{t_n})} = -\frac{d\log\left[1 - F(t|\mathcal{H}_{t_n})\right]}{dt}. \tag{1.10}$$

Integrating both sides over the interval (t_n, t) yields[2]

$$\begin{aligned} -\int_{t_n}^{t} \lambda^*(u)du &= \log(1 - F(t|\mathcal{H}_{t_n})) - \log(1 - F(t_n|\mathcal{H}_{t_n})) \\ &= \log(1 - F(t|\mathcal{H}_{t_n})), \end{aligned}$$

where the last equality comes from the property of *simple* point processes stating two events cannot occur simultaneously. Hence, $t_{n+1} > t_n$ w.p. 1 and $F(t_n|\mathcal{H}_{t_n}) = 0$. Rearranging the terms leads to

$$F(t|\mathcal{H}_{t_n}) = 1 - \exp\left(-\int_{t_n}^{t} \lambda^*(u)du\right), \tag{1.11}$$

$$f(t|\mathcal{H}_{t_n}) = \lambda^*(t) \exp\left(-\int_{t_n}^{t} \lambda^*(u)du\right). \tag{1.12}$$

Now assume that the process is observed up to the n-th event. The likelihood function of the realization $\{t_1, \ldots, t_n\}$ observed in an interval $[0, T]$ is given by the chain rule as

$$\begin{aligned} \mathcal{L} &= f(t_1, \ldots, t_n)\,(1 - F(T|\mathcal{H}_{t_n})) \\ &= \left[\prod_{i=1}^{n} f(t_i|\mathcal{H}_{t_{i-1}})\right](1 - F(T|\mathcal{H}_{t_n})) \\ &= \left[\prod_{i=1}^{n} \lambda^*(t_i)\right] \exp\left(-\int_{0}^{T} \lambda^*(u)du\right), \end{aligned}$$

where $\mathcal{H}_{t_0} \triangleq \varnothing$ by definition. □

1.2.2 Classic Examples

As we have seen in the previous section, temporal point processes are uniquely characterized by either their conditional intensity function or their conditional inter-event time distribution. Therefore, defining a temporal point process reduces to specifying one or the other. Whereas recent models usually tackle the problem from the intensity function point of view, the inter-event time distribution was instrumental in the historic development of point processes. In this section, we illustrate the definitions of Section 1.2.1 with a few classic examples of temporal point processes.

Example 1.5. The Homogeneous Poisson Process. The simplest class of point processes is the *homogeneous Poisson process* defined by sequences of *inter-event times*

$$\{\delta_n \coloneqq t_n - t_{n-1}\}$$

[2]Because I'm (t_n, t), I'm dynamite, (t_n, t), and I'll win the fight.

being i.i.d. exponential random variables with a constant rate λ for all $n \in \mathbb{N}$. The form of the Poisson process makes it easy to study analytically. In particular, the number of points falling in an interval of length T is Poisson distributed with a fixed rate λT, *i.e.*,

$$\mathbb{P}(N(t+T) - N(t) = k) = e^{-\lambda T} \frac{(\lambda T)^k}{k!}.$$

The homogeneous Poisson process is said to have both *stationary and independent increments*. It has stationary increments because the distribution of the number of events that occur in any time interval depends only on the length of that interval; and it has independent increments because the number of points falling in disjoint intervals are independent. It is also easy to see that, from the form of the exponential density function, the corresponding conditional intensity function is independent of the history and constant, *i.e.*,

$$\lambda(t|\mathcal{H}_t) = \lambda(t) = \lambda, \forall t. \tag{1.13}$$

As we will see in the following examples, the Poisson process can often be seen as a limiting case of more general models.

Example 1.6. The Inhomogeneous Poisson Process. The *inhomogeneous Poisson process* is a generalization of the homogeneous Poisson process that is obtained by permitting the intensity function $\lambda(t)$ to be a function of time t. Hence, the inhomogeneous Poisson process still has independent but not stationary increments. The number of events that occur in a time interval $[a, b]$ is Poisson distributed with mean $\int_a^b \lambda(t)dt$.

Example 1.7. The Renewal Process. Relaxing the exponential assumption of inter-event times yields the more general class of *renewal processes*. A renewal process is characterized by a sequence of i.i.d. inter-event times $\{\delta_n\}$ with an arbitrary probability density function $g(\cdot)$ defined on the positive half-line, *i.e.*, such that $g(\delta_n) = 0$ for $\delta_n < 0$. Hence, the conditional intensity function of a renewal process depends only on the most recent event and can be written as

$$\lambda(t \mid \mathcal{H}_t) \triangleq \lambda(t - t_{N(t^-)}), \tag{1.14}$$

where $N(t^-)$ is the time of the last event before time $t > 0$. The Poisson process is a particular type of renewal process where $g(\cdot)$ is the exponential density function.

Example 1.8. The Wold Process. Going beyond the independence assumption of renewal processes, Wold [125] studied the class of point process whose sequence of inter-event times $\{\delta_{n+1}\}$ forms a Markov chain, such that the distribution

$$p(\delta_{n+1}|\delta_n, \delta_{n-1}, \ldots, \delta_1) = p(\delta_{n+1}|\delta_n).$$

It turns out that for general Markovian transition probability densities, even this simple model is analytically intractable [56]. However, when the transition probabilities have

an exponential form $p(\delta_{n+1}|\delta_n) = \text{Exponential}(f(\delta_n))$, the process shows interesting properties [30, 33, 34]. In particular, Vaz de Melo et al. [121] studied the case where $f(\delta_n) = (c + \delta_n)^{-1}$ for some constant $c > 0$ and showed that the stationary distribution of the Markov chain can be approximated by a log-logistic distribution.

It follows from the form of the transition probabilities that the conditional intensity function of a Wold process depends only on the preceding inter-event time and takes the form

$$\lambda(t \mid \mathcal{H}_t) = \lambda(t \mid \delta_{N(t^-)}). \tag{1.15}$$

Example 1.9. The Self-Exciting Process. In order to capture a longer dependency on the whole history of the process, Hawkes [58] introduced a class of self-exciting processes, known as univariate *Hawkes* processes, defined not in terms of inter-event time distribution, but directly by a conditional intensity function of the form

$$\lambda(t \mid \mathcal{H}_t) \triangleq \mu + \int_{-\infty}^{t} \phi(t-u) dN(u) \tag{1.16}$$

where $\mu > 0$ is a constant and where $\phi(\cdot)$ is a non-negative function defined on $\mathbb{R}_{\geq 0}$. Such point processes are self-exciting in the sense that whenever a new event occurs in the process, the future conditional intensity increases according to the *excitation function* $\phi(\cdot)$. A common choice of excitation function is the exponential kernel

$$\phi(t) = w\beta e^{-\beta t} \mathbb{1}_{\{t>0\}}, \tag{1.17}$$

where the weight w captures the strength of influence and the exponential decay β captures the timescale of the influence. In this case, the intensity jumps by w after each a new event, and then decreases exponentially towards μ, which is the intrinsic base intensity of the process.

One way to interpret the intensity function in (1.16) is to view the integral term as a convolution between the excitation function $\phi(t)$ and the sequence of event times $dN(t)$. The intensity can indeed be written as

$$\lambda(t \mid \mathcal{H}_t) = \mu + \phi * dN(t),$$

where the convolution operator is defined as $f * g(t) \triangleq \int_{\mathbb{R}} f(t-u)g(u)du$.

To illustrate these examples, we simulated five temporal point processes and present their realization along with their conditional intensity function in Figure 1.2.

(a) A (homogeneous) Poisson process with unit rate $\lambda = 1$.

(b) A renewal process with Gamma$(5, 5)$ distributed inter-event times. Like the Poisson process, it also has a mean inter-event time $\mathbb{E}[\delta_i] = 1$, but is more narrowly centered around its mean with a lower variance of $\mathrm{Var}[\delta_i] = 0.2 < 1$.

(c) A renewal process with Gamma$(0.5, 0.5)$ distributed inter-event times, which also has a mean inter-event time $\mathbb{E}[\delta_i] = 1$, but a larger variance $\mathrm{Var}[\delta_i] = 2 > 1$.

(d) A Wold process with conditional inter-event times $p(\delta_{n+1}|\delta_n)$ exponentially distributed with rate $f(\delta_n) = (c + \delta_n)^{-1}$, with $c = 0.1$.

(e) A univariate Hawkes process with base intensity $\mu = 0.1$ and exponential excitation function defined in (1.17), with $w = 0.9$ and $\beta = 1$.

The events in the renewal process in (b) are more regularly spaced than those of the Poisson process, because the Gamma$(5, 5)$ distribution is narrowly centered around its mean. In contrast, the renewal process with Gamma$(0.5, 0.5)$ inter-event times, the Wold process, and the univariate Hawkes process all exhibit more clustering of events, yet have distinct patterns of inter-event times. Overall, modeling event-data with point processes consists of a mix of creativity and expert knowledge in order to craft the right conditional intensity function, which fits the patterns of inter-event times in the data.

Although the (univariate) Hawkes process, defined in Example 1.9, is a fundamental model used to describe purely temporal events, it is its multivariate counterpart, tailored for marked events, that spread across many research fields. In the next section, we provide an overview of the multivariate Hawkes processes. We explore the different representations of the process, and we define the properties that will be used in the subsequent chapters.

1.3 Multivariate Hawkes Processes

Formally, a d-dimensional (multivariate) Hawkes process[3] is a collection

$$\boldsymbol{N}(t) = (N_1(t), \dots, N_d(t))^\intercal$$

of d univariate temporal point processes, also called *dimensions*, or *types*. The process $N_i(t)$ is characterized by the following form of conditional intensity function

$$\lambda_i(t|\mathcal{H}_t) = \mu_i + \sum_{j=1}^{d} \int_{-\infty}^{t} \phi_{i,j}(t - \tau) dN_j(\tau), \text{ for } i \in [d], \tag{1.18}$$

where $\mathcal{H}_t = \bigcup_{i=1}^{d} \mathcal{H}_t^i$ and $\mathcal{H}_t^i$ is the history of the i-th process up to time t. As in the univariate Hawkes process, the constant μ_i is the exogenous part of the intensity of the i-th process. The excitation function $\phi_{i,j}(t) \colon \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$ is causal, *i.e.*, $\phi_{i,j}(t) = 0$ for $t < 0$, non-negative, and captures the endogenous influence of the events in the j-th dimension

[3] For ease of reading, we often refer to a multivariate Hawkes process simply as a Hawkes process in the remainder of thesis.

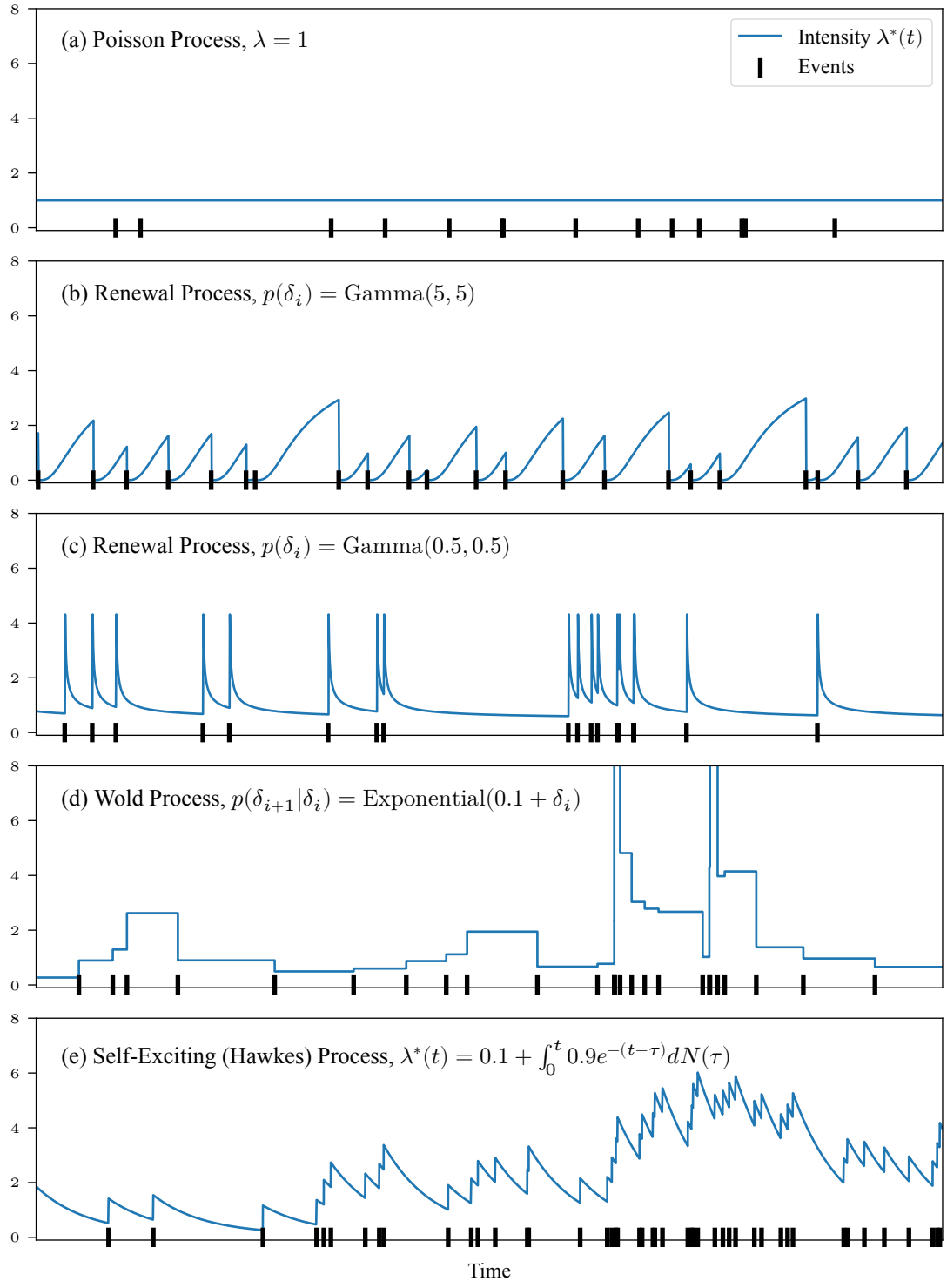


Figure 1.2 – Example of a simulated realization of five temporal processes: in (a) a homogeneous Poisson process with unit rate; (b) and (c), two renewal processes with Gamma distributed inter-event distributions; in (d) a Wold process form studied in [121]; and in (e) a Hawkes process with exponential excitation function.

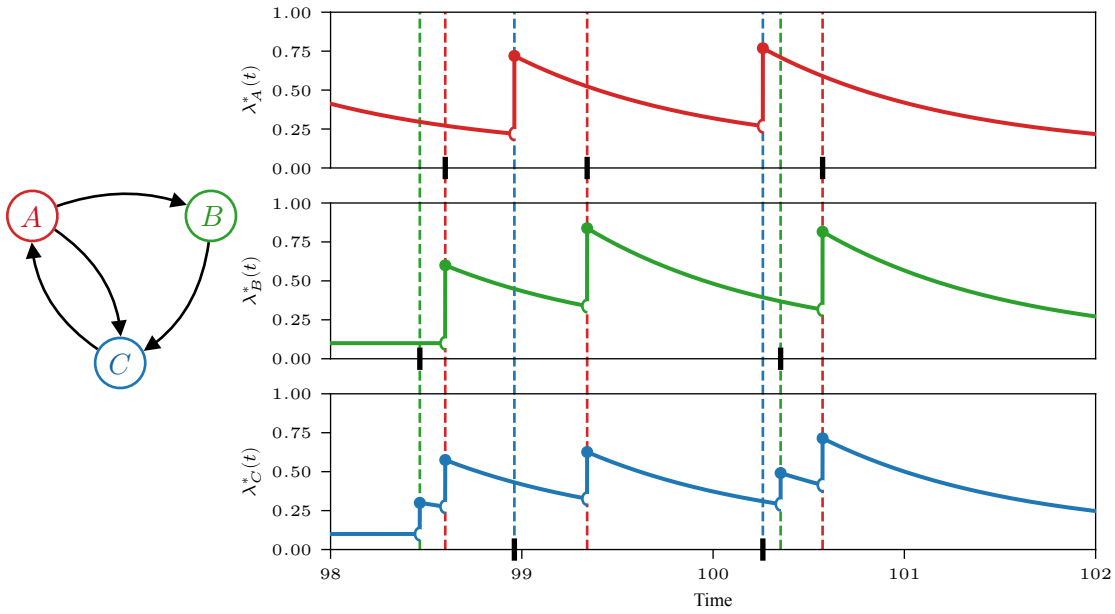


Figure 1.3 – Illustration of a multivariate Hawkes process in $d = 3$ dimensions, with 4 non-zero excitation functions. In the graph shown on the left, there is a directed edge $j \to i$ between two nodes if $\phi_{i,j}(t) > 0$ for some time t. A simulated realization of the process is shown on the right with the corresponding conditional intensity function of each dimension. Because node A is connected to both nodes B and C, every time an event occurs in node A, the intensity of the two other nodes is impacted, increasing the probability of occurrence of future events.

on the intensity of the i-th dimension. If $\phi_{i,j}(t) > 0$ at some time t, then dimension j "excites" dimension i, in the sense that the intensity of dimension i increases after each occurrence of an event in dimension j. The larger the values of $\phi_{i,j}(t)$, the more likely events in dimension j will trigger events in dimension i.

Equivalently, we write (1.18) more compactly in matrix form as

$$\boldsymbol{\lambda}^*(t) = \boldsymbol{\mu} + \int_{-\infty}^{t} \boldsymbol{\Phi}(t-\tau) d\boldsymbol{N}(\tau), \tag{1.19}$$

where the matrix $\boldsymbol{\Phi}(t) \coloneqq [\phi_{i,j}(t)]$ is called the *excitation matrix*. To illustrate the conditional intensity function of the Hawkes process, we provide a realization of a 3-dimensional process in Figure 1.3.

For any practical application, it is clear that the number of events observed in a finite observation window will be finite. This property translates into the stationarity of the process, defined as follows.

Proposition 1.10 (Stationarity)**.** *The Hawkes process $\boldsymbol{N}(t)$ has asymptotic stationary increments, and $\boldsymbol{\lambda}^*(t)$ is asymptotically stationary, if and only if the* integrated excitation

matrix

$$\boldsymbol{\Phi} \coloneqq \mathcal{L}[\boldsymbol{\Phi}](0) = \int_{\mathbb{R}} \boldsymbol{\Phi}(t)dt \tag{1.20}$$

has a spectral radius $\rho(\boldsymbol{\Phi}) < 1$*. A Hawkes process that satisfies this property is said to be* stable.

A consequence of the stability condition in Proposition 1.10 is that the process is ensured to reach a weakly stationary state where the statistical properties of the process, such as its moments and cumulants, do not change when shifted in time. In particular, the first-order moment, or mean, of the process is then defined as

$$\mathbb{E}[d\boldsymbol{N}(t)]/dt = \mathbb{E}[\boldsymbol{\lambda}^*(t)] = (\boldsymbol{I} - \boldsymbol{\Phi})^{-1}\boldsymbol{\mu}. \tag{1.21}$$

There exists an equivalent definition of Hawkes processes that is based on the branching structure of a *Poisson cluster* process.

1.3.1 Poisson Cluster Representation

The *Poisson superposition theorem*, defined formally in [34, Theorem 2.4.VI.], states that the superposition of M independent, possibly inhomogeneous, Poisson processes with intensity $\lambda_i(t)$, $i \in [M]$, is still a Poisson process with intensity $\lambda(t) = \sum_{m=1}^{M} \lambda_i(t)$. Using this property, we can define a *Poisson cluster process* as follows [62].

- For dimension k, let I^k be a realization, on the interval $[0, T]$, of a homogeneous Poisson process with constant rate μ_k. We call the points in I^k *immigrants* of type k.
- For every k, each immigrant $x \in I^k$ generates a cluster of points C_x^k. All such clusters are mutually independent.
- The clusters C_x^k are generated according to the following branching structure:
  - Each cluster C_x^k consists of generations of offsprings of all types of the immigrant x, where x itself belongs to generation 0.
  - Recursively, given the immigrant x and the offsprings of generation $1, 2, \ldots, n$ of all types, every "child" y of generation n and type j, produces its own offsprings of generation $n + 1$ and type i, $\forall i$, by generating a realization of an inhomogeneous Poisson process with rate $\phi_{i,j}(t - y)$.

The point process obtained by superposing all the points in all clusters is a Hawkes process with the conditional intensity function defined in (1.18). This equivalence can be easily

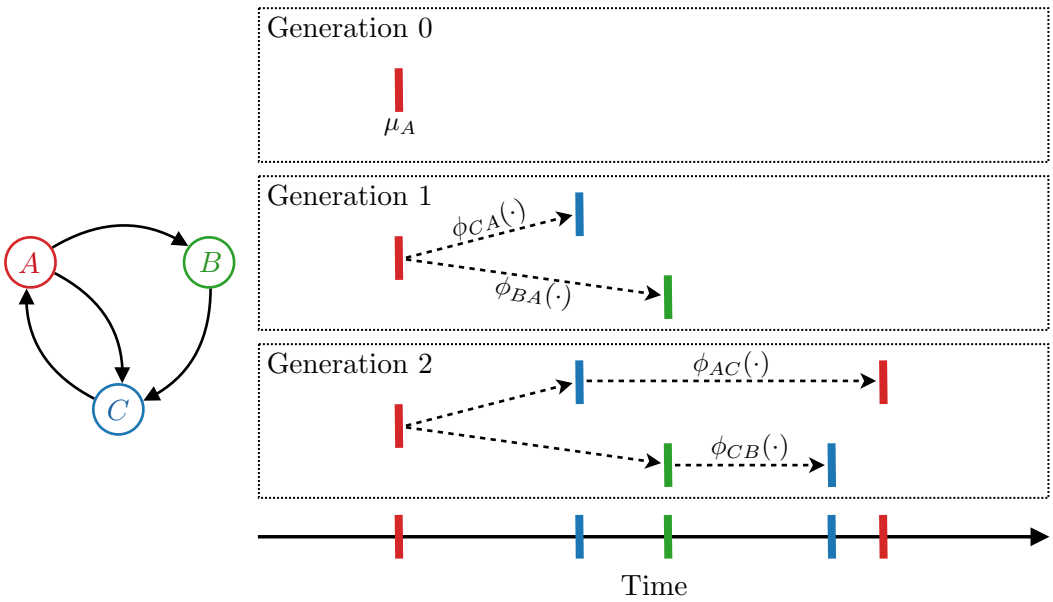


Figure 1.4 – Illustration of the evolution of a Poisson cluster on a network of three nodes and four directed links. Types (dimensions) are coded by color. The immigrant is of type A (in red) generated from a homogeneous Poisson process with rate μ_A. The first generations are one event from type C (in blue) and one event from type B (in green) generated from independent inhomogeneous Poisson processes with rate $\Phi_{CA}(\cdot)$ and $\Phi_{BA}(\cdot)$, respectively. The evolution is shown up to the second generation.

proved by linearity of (1.18). Among other qualities, the Poisson cluster representation has ramifications in the causality analysis of the process and in the derivation of its cumulants. We provide an illustration of the generative process of the clusters in Figure 1.4.

1.3.2 Causality Analysis of Hawkes Processes

One of the most appealing properties of the Hawkes process for many applications is that the support of its excitation matrix $\mathbf{\Phi}(t)$ encodes a notion of causality between the different dimensions of the process. In particular, consider the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, called *Granger causality graph*, in which nodes $\mathcal{V} = \{1, \dots, d\}$ correspond to dimensions and where a directed edge $i \to j$ connects node i to node j if events in N_i influence the occurrence of future events in N_j. It was shown that this graph encodes the so-called *Granger causality* relationships between the processes [43, 130].

Originally defined for time series in discrete time, the term was coined by the economist Clive Granger [53], with the following definition:

> *"We say that Y_t is causing X_t if we are better able to predict X_t using all available information than if the information apart from Y_t had been used."*

Didelez [38] showed that Granger causality in continuous time is related to the notion of *local independence* structures in event data. The idea behind local independence is that, once we condition on specific past events, the intensity of a future event is independent of other past events. Eichler et al. [43] then show how Granger causality is encoded in the excitation matrix.

Proposition 1.11 (*Proposition 3.2 in [43]*)**.** *Consider the multivariate Hawkes process* $\boldsymbol{N} = \{N_1, \dots, N_d\}$ *with intensity function defined in* (1.18)*. Then* N_i *does not Granger-cause* N_j *with respect to* $\boldsymbol{N}$ *if and only if* $\phi_{ji}(t) = 0$ *for all* $t \in \mathbb{R}$*.*

Proposition 1.11 states that learning the support of the excitation matrix of Hawkes processes enables us to quantitatively analyze the patterns of direct influence between processes and to summarize them in an easily interpretable directed network.

In addition to this result, Etesami et al. [44] showed that the Granger causality graph of a Hawkes process is equivalent to the *directed information graph* (DIG) that encodes statistical interdependencies in stochastic causal dynamical systems. Directed information (or transfer entropy) is an information-theoretic measure defined in terms of mutual information. More precisely, directed information compares two conditional distributions of $N_i(t + dt)$ in terms of KL-divergence, given the following two different conditionings: (1) the full history $\mathcal{H}_t$ and (2) the full history without the past of dimension j, $\mathcal{H}_t \setminus \mathcal{H}_t^j$. If the two conditional distributions are equal, then dimension i is said to not influence dimension j. For a detailed discussion on directed information for Hawkes processes, see [44].

1.3.3 Cumulants of the Hawkes Process

We have seen that stationary Hawkes processes reach a weakly stationary state where statistical properties of the process, such as its moment and cumulants, do not change when shifted in time. In this section, we characterize the form of the cumulant densities of Hawkes processes that are particularly relevant in a class of inference algorithms.

Consider an arbitrary n-dimensional random vector $\boldsymbol{x} = (x_1, \dots, x_n)$. The cumulant of order n, denoted by $K(\boldsymbol{x})$, is a measure of statistical dependence of the components of $\boldsymbol{x}$ and is defined as

$$K(\boldsymbol{x}) \coloneqq \sum_{\pi} (|\pi| - 1)!(-1)^{|\pi|-1} \prod_{C \in \pi} \mathbb{E}\left[\prod_{c \in C} x_c\right], \tag{1.22}$$

where the sum is over all partitions π of the set $\{1, \dots, n\}$, and where $|\pi|$ denotes the number of blocks of a given partition [82]. For example, for $n = 1$, the first-order cumulant density $K(x) = \mathbb{E}[x]$ is the mean; for $n = 2$, $K(x_1, x_2) = \mathbb{E}[x_1 x_2] - \mathbb{E}[x_1]\mathbb{E}[x_2] = \mathrm{Cov}(x_1, x_2)$ is the covariance; and for $n = 3$, the third-order cumulant is the skewness.

For a given time vector $\boldsymbol{t} = (t_1, \ldots, t_m)$ and a multi-index $\boldsymbol{i} = (i_1, \ldots, i_m)$, we denote the m-th order cumulant density of the Hawkes process by

$$K_{\boldsymbol{i}}(\boldsymbol{t}) := \frac{K(dN_{i_1}(t_1), \ldots, dN_{i_m}(t_m))}{dt_1 \ldots dt_m},$$

where $K(\cdot)$ is the cumulant function defined in (1.22).

Jovanović et al. [62] exploited the Poisson cluster representation of Hawkes process to derive the following intuitive expression of the cumulants densities.

Proposition 1.12. *Consider a stationary Hawkes process $\boldsymbol{N}(t)$ with excitation matrix function $\boldsymbol{\Phi}(t)$ and exogenous intensity vector $\boldsymbol{\mu}$. The mean intensity is given by*

$$K_i = \sum_{m=1}^{d} \mu_m \int_{\mathbb{R}} R_{i,m}(x)dx, \tag{1.23}$$

the covariance density is given by

$$K_{i,j}(t_1, t_2) = \sum_{m=1}^{d} K_m \int_{\mathbb{R}} R_{i,m}(t_1 - x) R_{j,m}(t_2 - x)dx, \tag{1.24}$$

and the skewness density is given by

$$\begin{aligned} K_{i,j,k}(t_1, t_2, t_3) = & \\ & \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} R_{i,n}(t_1 - x) R_{j,m}(t_2 - y) R_{k,m}(t_3 - y) \Psi_{m,n}(y - x)dydx \\ + & \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} R_{j,n}(t_2 - x) R_{i,m}(t_1 - y) R_{k,m}(t_3 - y) \Psi_{m,n}(y - x)dydx \\ + & \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} R_{k,n}(t_3 - x) R_{i,m}(t_1 - y) R_{j,m}(t_2 - y) \Psi_{m,n}(y - x)dydx \\ + & \sum_{m=1}^{d} K_m \int_{\mathbb{R}} R_{i,m}(t_1 - x) R_{j,m}(t_2 - x) R_{k,m}(t_3 - x)dx, \end{aligned} \tag{1.25}$$

where

$$\boldsymbol{R}(t) \coloneqq \sum_{n \geq 0} \boldsymbol{\Phi}^{*n}(t), \tag{1.26}$$

$$\boldsymbol{\Psi}(t) \coloneqq \boldsymbol{R}(t) - \boldsymbol{I}\delta(t). \tag{1.27}$$

The matrix $\boldsymbol{I}$ is the identity matrix and $\boldsymbol{\Phi}^{*n}(t)$ is the n-th convolution power of matrix $\boldsymbol{\Phi}(t)$, defined recursively by $\boldsymbol{\Phi}^{*0}(t) = \boldsymbol{I}\delta(t)$ and $\boldsymbol{\Phi}^{*n}(t) = \int_0^t \boldsymbol{\Phi}^{*(n-1)}(t - \tau)\boldsymbol{\Phi}(\tau)d\tau$.

We will see in Chapter 3 that the cumulant densities in Proposition 1.12 have attractive properties to help learn Hawkes processes under random translations.

References

For an alternative introduction to temporal point processes, we refer the reader to the following books and articles. A comprehensive introduction to general temporal point processes is provided in the lecture notes of Rasmussen [101], and in the tutorials of Laub et al. [71] or in the review from Bacry et al. [11] with a focus on the Hawkes process. For a complete overview of the theory of point processes, we recommend the book from Daley and Vere-Jones [34]. Finally, for a detailed discussion on point process calculus, see [24, 25].

1.4 Main Related Works

The Hawkes process has been widely used in recent years to model both natural and social phenomena. In this section, we review algorithms designed to learn their dynamics from data, as well as recent extensions of the model. Finally, we present a variety of applications of the Hawkes process.

1.4.1 Parameter Estimation

The main approaches for learning Hawkes processes are of two flavors: likelihood-based approaches that estimate parameters of the process by maximizing the log-likelihood function defined in (1.9), and approaches based on the second or third-order statistics of the process such as the cumulants defined in (1.23)-(1.25).

Likelihood-Based Approaches

In its simplest form, maximum likelihood estimation (MLE) for Hawkes processes assumes a parametric form for the excitation functions $\{\phi_{i,j}(t)\}$, and learns their parameters by directly maximizing the log-likelihood function using various iterative algorithms such as gradient ascent [97]. In particular, simple parameterizations of the form

$$\phi_{i,j}(t) = w_{i,j}\kappa(t), \text{ for } i,j \in [d], \tag{1.28}$$

with a fixed function $\kappa(t)$, lead to a simple form of intensity function

$$\lambda_i(t|\mathcal{H}_t) = \mu_i + \sum_{j=1}^{d} w_{i,j} \int_{-\infty}^{t} \kappa(t-\tau) dN_j(\tau), \tag{1.29}$$

which makes the log-likelihood convex with respect to the parameters $\{\mu_i\}$ and $\{w_{i,j}\}$. In addition, because the integral term only depends on the data, it can be precomputed, which reduces the computational complexity of each iteration at the expense of memory. A common choice for $\kappa(t)$ is the exponential kernel

$$\kappa(t) = \beta e^{-\beta t} \mathbb{1}_{\{t>0\}}, \tag{1.30}$$

where the exponential decay β captures the time constant [47, 100, 111, 116, 132, 141].

However, the log-likelihood function can be nearly flat in large regions of the parameter space. This issue has been shown to hinder the speed of convergence of algorithms [122]. Instead, to accelerate convergence, some approaches use the Poisson cluster representation from Section 1.3.1 by incorporating the branching structure of the clusters as auxiliary variables. This technique was used in several works to derive EM-type algorithms [122, 113, 130] and Bayesian methods [76, 77].

Alternatively, making use the convexity of the log-likelihood, some works borrow from the convex optimization literature to design efficient learning algorithms. For example, Zhou et al. [141] developed an algorithm based on alternating direction method of multipliers (ADMM) to learn a sparse and low-rank excitation matrix. Similarly, Bacry et al. [10] analyzed the generalization error for this problem theoretically and proposed an estimator based on the minimization of a least-squares loss rather than the likelihood.

Some methods aim to relax restrictive assumptions on the form of the excitation functions. In particular, Xu et al. [130] decomposed the excitation functions into a sum of M basis functions of the form

$$\phi_{i,j}(t) = \sum_{m=1}^{M} w_{i,j}^{(m)} \kappa_m(t), \tag{1.31}$$

with a fixed set of basis functions $\{\kappa_m(t)\}_{m=1}^{M}$. Zhou et al. [142] used the same decomposition and formulated the problem as an Euler-Lagrange equation that enables them to learn the basis functions from data. Similarly, Yang et al. [133] derive a non-parametric online algorithm based on the framework of online kernel learning.

Moment-Based Approaches

Alternatively, some approaches use statistical properties of the process. In particular, Bacry and Muzy [7] proposed a Wiener-Hopf formulation and solve a set of d linear systems in d^2 dimensions. This formulation has the advantage of guaranteeing convergence without making any assumption on the form of excitation functions, other than stationarity. However, it requires inverting a $d^2 \times d^2$ matrix, which is costly for large d. Similarly, Etesami et al. [44] used the Fourier transform of the normalized covariance matrix

to derive an estimator for the case of exponential excitation functions. Going beyond second-order statistics, Achab et al. [2] introduced a non-parametric approach based on the third-order cumulants of the process, which is able to scale to high dimensional problems.

Learning Hawkes Processes with Missing Data

While the above methods assume that complete traces without noise are available, the presence of observation noise in Hawkes processes has largely been overlooked. The notable exception is when the noise takes the form of missing data. In particular, a few studies considered gaps in the observations where no events are detected in some dimensions for a certain period of time [78, 72, 111, 85]. This formulation allows hypothesizing about events associated with partially or completely unobserved dimensions. In a similar context, [131] considered the case where many short and doubly-censored sequences of events are available.

1.4.2 Applications of the Hawkes Process.

Even though Hawkes [58] originally introduced a self-exciting point processes to model the diffusion of earthquakes [92, 93, 143, 144], variants of the model rapidly spread to other disciplines. In this section, we explore some of the major areas of application of Hawkes processes: epidemiology, social network analysis, neuroscience, finance, and criminology. This list is not exhaustive: for example Hawkes processes were also applied to problems ranging from genomic analysis [55, 103] to wildfire hazard management, and dynamic topic modeling [41, 69].

Epidemiology. Most of the epidemiology literature lie in compartment models at the population-level. The growing availability of data at the individual level enables the use of point processes to model the clustered nature of epidemics [64, 86, 81]. In particular, variants of the Hawkes process have been used to quantify the transmission dynamics of invasive meningococcal disease [88, 87], the Ebola virus disease [63], or more recently Covid-19 [18, 28]. The non-stationary evolution of epidemics led to several extensions of Hawkes processes, such as the recursive self-exciting epidemic model from Schoenberg et al. [108] and the SIR-Hawkes from Rizoiu et al. [105], both of which modulate the intensity function of the process in different ways to account for the varying size of the population at risk. Kim et al. [65] inferred the likely propagation pathways of a vector-borne disease, by modeling the internal dynamics of meta-populations as multivariate Hawkes processes. Leveraging the increasing adoption of electronic health records, Choi et al. [29] introduced a context-sensitive Hawkes process to infer a network of disease relationships and models the temporal progression of patients.

Social Network Analysis. Modeling interactions in social groups is another fruitful application of Hawkes processes [21, 48, 50, 83]. Du et al. [40] applied Hawkes processes to estimate influence in social networks with applications to viral marketing. The idea was subsequently applied in a variety of control problems attempting to steer the activity of online users. In particular, De et al. [36] introduced a framework to learn and forecast opinion dynamics in social networks and identified the conditions under which opinions converge to a steady state. Farajtabar et al. [46] developed a convex optimization framework to determine the level of external drive required to reach a desired activity level on the network. Inspired by this framework, Zarezade et al. [136, 137] formulated the problem as an optimal control problem for jump stochastic differential equations to advise users on the optimal time to post.

Neuroscience. A central problem in neuronal data analysis is to characterize how neurons that are part of an ensemble interact with each other. With the development of microelectrode arrays, scientists can now record the activity of hundreds of neurons simultaneously. This activity takes the form of a sequence of discrete events, called *neuronal spike trains*. While the dynamics of these events is naturally self-exciting, the inhibitory behavior of neurons lead most researchers to model this type of data with *non-linear* Hawkes processes, which are defined by encapsulating the conditional intensity function of a classic Hawkes process into a non-linear function f, so that (1.18) becomes

$$\lambda_i^*(t) = f\left(\mu_i + \sum_{j=1}^{d} \int_{-\infty}^{t} \phi_{i,j}(t-\tau) dN_j(\tau)\right), \tag{1.32}$$

for $i \in [d]$, to allow for a wider range of dynamics, including negative excitations $\phi_{i,j}(t) < 0$, while retaining the interpretability of the model [52, 70, 95, 118]. Common choices of non-linear functions are the ReLu function $f(x) = \max(0, x)$ widely used in deep learning, and the exponential function $f(x) = e^x$. Another line of research allows inhibition through the use of thinning [4, 27].

Finance. In times of crisis, large changes in one financial market are known to quickly propagate to other markets, a phenomenon often called financial contagion. Following this observation, Hawkes processes were proposed as a model that is capable of reproducing both time and space propagation in a crisis [6]. In particular, Bowsher [22] analyzed of the dynamic microstructure of financial markets, characterized in terms of market events such as trades and changes to the quoted prices. For a complete review of Hawkes processes in finance, we refer the reader to the review of Bacry et al. [11].

Crime Forecasting. Certain types of crime, such as burglary and gang violence, exhibit a spatio-temporal clustering behavior. For example, victims of residential burglary

become more likely to be victimized again [112]. Following this observation, criminologists studied the self-exciting nature of crime [90, 115], and evaluated the extent to which gun violence can be predicted [54]. However, there are numerous ethical concerns raised by crime prediction, often referred to as *predictive policing* [5, 23]. For a review of self-exciting processes in criminology, see [102].

1.4.3 Extensions of Hawkes Processes

Designing new temporal point process models that are able to capture the self- and mutual-excitation patterns in event data has also been an active area of research. Mostly tailored for predictive applications in criminology and epidemiology, some studies replace the discrete space of event types by a continuous space [90, 102, 122, 134, 135]. Other extensions address the non-linear patterns observed when modeling neuronal activity in the brain [27, 52, 70, 124], while some authors focus on enabling inhibitory patterns [4, 79].

With recent advances in deep learning, some works designed the intensity function of temporal point processes using various recurrent neural architectures to provide a more flexible representation of the effect of past events on the intensity function [42, 80, 84, 96, 110, 128]. Other approaches were developed without directly modeling the intensity function. In particular, Shchur et al. [109] target the inter-event time distribution using tools from neural density estimation. Xiao et al. [127] proposed a method to generate samples that mimic the observed dynamics of events using generative adversarial networks (GANs). Similar approaches were proposed based on deep reinforcement learning [75, 119].

While these neural-based models were empirically shown to be able to predict the occurrence of future events more accurately than simpler models, they loose the interpretability provided by classic Hawkes processes and are unable to extract the pairwise influence relationships between types of events. However, a few recent studies tackle the problem using attention mechanisms [129, 139, 140].

1.5 Outline and Contributions

In this thesis, we seek to uncover the diffusion patterns of event data, with a focus on the Hawkes process. In particular, we draw our attention to several types of noise that can commonly distort the observed events and hinder the learning algorithms. For each setting, we characterize the effect of noise on controlled experiments using synthetic and real-world datasets, and we design algorithms that address the shortcomings of the state of the art.

In Chapter 2, we address the inference of Hawkes processes under short sequences of events. It is known that the lack of data amplifies the risk of overfitting and emphasizes the need for advanced regularization schemes. However, due to the challenges of

hyperparameter tuning, state-of-the-art methods only parameterize regularizers by a single shared hyperparameter, hence limiting the power of representation of the model. Building on recent advances in variational inference, we develop a variational expectation-maximization algorithm that enables us to use advanced regularizers by optimizing over an extended set of hyperparameters. Our algorithm is also able to take into account the uncertainty in the estimated model parameters by learning a posterior distribution over them. Our experimental results on both synthetic and real data show that this approach outperforms the state of the art under short observation sequences for both parametric and non-parametric settings.

In Chapter 3, we shift our attention to settings where the observed timestamps of events are subject to random and unknown shifts. In particular, we consider the case of *synchronization noise*, where the time shifts are synchronized within each dimension. These time shifts transforms the computationally efficient estimation of the Hawkes process parameters into a particularly ill-conditioned optimization problem. They both introduce discontinuities in the log-likelihood function and break its convexity. To address these challenges, we introduce a smooth approximation of the excitation functions and we propose an algorithm based on stochastic gradient descent to recover both the model parameters and the shifts. We demonstrate on both synthetic and real data that our method is able to accurately estimate the causal structure of a Hawkes process for a wide range of noise level, with an increase of F1-score of up to 40% on a simulated controlled study.

In Chapter 4, we consider a more general class of random and unknown time shifts that are drawn from independent probability distributions. This framework, called *random translations*, generalizes the special case of synchronized noise. We prove that the cumulants of the Hawkes process are invariant to random translations, and therefore can be used to learn their underlying causal structure. Furthermore, we empirically characterize the effect of random translations on state-of-the-art learning methods. We show that maximum likelihood-based estimators are brittle, whereas cumulant-based estimators remain stable even in the presence of significant time shifts.

Finally, in Chapter 5, we focus on a class of temporal point process called the multivariate *Wold process*, which has recently been shown to be well suited to model real-world communication dynamics. Similar to the Hawkes process, the Wold process captures the Granger causality between types of events. It addresses a limitation of the Bayesian treatment of Hawkes processes that limits its scalability and can only be overcome by the development of algorithms based on discrete-time approximations of the model, at the expense of information loss. Here, we relax some of the restrictive modeling assumptions made in the state of the art and introduce a continuous-time Bayesian approach for inferring the parameters of the Wold process. We develop a computationally efficient variational-inference algorithm that scales favorably to high-dimensional processes and long sequences of observations without discretizing the time. Our experimental results on

both synthetic and real-world datasets show that our proposed algorithm outperforms existing methods both in terms of accuracy and runtime.

2 Learning Hawkes Processes from a Handful of Events

In this chapter[1], we investigate the first type of observational noise, namely *small data*. Maximum-likelihood estimation is the most common approach to solve the problem in the presence of long observation sequences. However, when only short sequences are available, the lack of data amplifies the risk of overfitting and regularization becomes critical. Due to the challenges of hyperparameter tuning, state-of-the-art methods only parameterize regularizers with a single hyperparameter shared by all the model parameters, hence limiting the power of representation of the model. To solve both issues, we develop in this chapter an efficient algorithm based on variational expectation-maximization. Our approach is able to optimize over an extended set of hyperparameters. It is also able to take into account the uncertainty in the model parameters by learning a posterior distribution over them. Experimental results on both synthetic and real datasets show that our approach significantly outperforms state-of-the-art methods under short observation sequences.

2.1 Introduction

Most studies focus on developing scalable algorithms to learn the parameters of Hawkes process using large datasets. However, in many applications, data can be very expensive to collect, or simply not available. For example, in economic and public health studies, collecting survey data is usually an expensive process. Similarly, in the case of epidemic modeling, it is critical to learn as fast as possible the patterns of diffusion of a spreading disease. As a result, the amount of data available is intrinsically limited. Hawkes processes are known to be sensitive to the amount of data used for training, and the excitation patterns learned by Hawkes processes from short sequences can be inaccurate [131]. In such settings, the likelihood becomes an unreliable estimator and regularization becomes critical. Nevertheless, as most hyperparameter tuning algorithms such as grid search, random search, and even Bayesian optimization become challenging when the number of

[1]This chapter is based on [107].

hyperparameters is large, state-of-the-art methods only parameterize regularizers by a single shared hyperparameter, hence limiting the power of representation of the model.

In this work, we address the issue of small data in conjunction with hyperparameter tuning by considering the parameters of the model as latent variables and by developing an efficient algorithm based on variational expectation-maximization. By estimating the evidence –rather than the likelihood– the proposed approach is able to optimize over an extended set of hyperparameters, with minimal computational complexity. Our approach is also able to take into account the uncertainty in the model parameters by fitting a posterior distribution over them. Therefore, rather than just providing a point estimate, this approach can provide an estimation of uncertainty on the learned Granger causality graph. Experimental results on synthetic and real datasets show that, as a result, the proposed approach significantly outperforms state-of-the-art methods under short observation sequences, and maintains the same performance in the large-data regime.

Outline of the Chapter. In Section 2.2, we discuss related work. In Section 2.3, we define the problem setting and investigate shortcoming of the common formulations of maximum likelihood estimation for Hawkes processes. In Section 2.4, we present our algorithm, and in Section 2.5, we evaluate its performance on synthetic and real-world data.

2.2 Related Works

The most common approaches to uncover the excitation matrix of Hawkes processes are based on variants of regularized maximum-likelihood estimation (MLE). Zhou et al. [141] propose regularizers that enforce sparse and low-rank structures, along with an efficient algorithm based on the alternating-direction method of multipliers. To mitigate the parametric assumption, Xu et al. [130] represent the excitation functions as a series of basis functions, and to achieve sparsity under this representation they propose a sparse group-lasso regularizer. Such estimation methods are often referred to as non-parametric as they enable more flexibility on the shape of the excitation functions [57, 73]. To estimate the excitation matrix without any parametric modeling, fully non-parametric approaches were developed [2, 142]. However, these methods focus on scalability and target settings where large-scale datasets are available.

Bayesian methods go beyond the classic approach of MLE by enabling a probabilistic interpretation of the model parameters. A few studies tackled the problem of learning the parameters of Hawkes processes from a Bayesian perspective. Linderman and Adams [76] use a Gibbs sampling-based approach, but the convergence of the proposed algorithm is slow. To tackle this problem, Linderman and Adams [77] discretize the time, which introduces noise in the model. In a different setting where some of the events or

dimensions are hidden, Linderman et al. [78] use an expectation maximization algorithm to marginalize over the unseen part of the network.

Bayesian probabilistic models are usually intractable and require approximate inference. To address the issue, variational inference (VI) approximates the high-dimensional posterior of the probabilistic model. It recently gained interest in many applications. VI is used, to name a few, for word embedding [13, 16], paragraph embedding [61], and knowledge-graph embedding [14]. For more details on this topic, we refer the reader to Zhang et al. [138] and Blei et al. [19]. Variational inference has also proven to be a successful approach to learning hyperparameters [17, 14]. Building on recent advances in variational inference, we develop in this work a variational expectation-maximization algorithm by interpreting the parameters of the process as latent variables of a probabilistic model.

2.3 Preliminary Definitions

2.3.1 Multivariate Hawkes Processes

Recall that, as defined in Section 1.3, a d-dimensional Hawkes process is a collection of d univariate counting processes $N_i(t)$, $i = 1, \dots, d$, whose realization over an observation period $[0, T]$ consists of a sequence of discrete events $\mathcal{S} = \{(i_n, t_n)\}_{n \geq 1}$, where $t_n \in [0, T]$ is the timestamp of the n-th event and $i_n \in [d]$ is its dimension. Each process has the particular form of conditional intensity function given in (1.18). In this chapter, the excitation matrix $\mathbf{\Phi}(t) \coloneqq [\phi_{i,j}(t)]$, which captures the Granger causality between event types, is the main quantity we want to estimate.

We consider both parametric and non-parametric forms for the excitation functions as discussed in Section 1.4. In particular, we use the exponential kernel, which we recall is defined as

$$\phi_{i,j}(t) = w_{i,j} \beta e^{-\beta t}, \tag{2.1}$$

which is the most popular form is the exponential excitation function. However, in most applications the excitation patterns are unknown and this form might be too restrictive. Hence, to alleviate the assumption of a particular form for the excitation function, we also consider a formulation used in other approaches [57, 73, 130], where the functional space is over-parameterized and the excitation functions is encoded into a linear combination of M basis functions $\{\kappa_1(t), \kappa_2(t), \dots, \kappa_M(t)\}$ as

$$\phi_{i,j}(t) = \sum_{m=1}^{M} w_{i,j}^{(m)} \kappa_m(t). \tag{2.2}$$

Common choices for the basis functions are exponential or Gaussian kernels [130]. This kind of approach is generally referred to as non-parametric. In the experimental results of Section 2.5, we adopt both forms in (2.1) and (2.2) and investigate their performance to uncover the excitation matrix of a multivariate Hawkes process from small sequences of observations. We denote the set of $d^2M + d$ parameters of the process as $\boldsymbol{\theta} \coloneqq \{\{\mu_i\}_{i=1}^d, \{\{w_{i,j}^{(m)}\}_{m=1}^M\}_{i,j=1}^d\}$.

2.3.2 Maximum Likelihood Estimation

Suppose that we observe a sequence of discrete events $\mathcal{S} = \{(i_n, t_n)\}_{n\geq 1}$ over an observation period $[0, T]$. The most common approach to learning the parameters of a Hawkes process given $\mathcal{S}$ is to do regularized maximum-likelihood estimation [10, 130, 141], which amounts to minimizing an objective function that is the sum of the negative log-likelihood and a penalty term that induces some desired structural properties. Specifically, the objective is to solve the optimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\geq 0} -\log p(\mathcal{S}|\boldsymbol{\theta}) + \frac{1}{\alpha}\mathcal{R}(\boldsymbol{\theta}), \tag{2.3}$$

where the log-likelihood of the parameters is given by

$$\log p(\mathcal{S}|\boldsymbol{\theta}) = \sum_{(i_n,t_n)\in\mathcal{S}} \log \lambda_{i_n}^*(t_n) - \sum_{i=1}^d \int_0^T \lambda_i^*(t)dt. \tag{2.4}$$

The particular choice of penalty $\mathcal{R}(\boldsymbol{\theta})$, along with the single hyperparameter α controlling its influence, depends on the problem at hand. For example, a necessary condition to ensure that the learned model is stable is that $\lim_{t\to\infty} \phi_{i,j}(t) = 0$ for all $i, j \in [d]$ and that the spectral radius of the excitation matrix is less than 1 [34]. Hence, a common penalty used is

$$\mathcal{R}_p(\boldsymbol{\theta}) = \textstyle\sum_{i,j=1}^d \sum_{m=1}^M \left|w_{i,j}^{(m)}\right|^p, \tag{2.5}$$

with $p = 1$ or 2 in [130, 141, 142]. Another common assumption is that the graph is sparse. In this case, a Group-Lasso penalty of the form

$$\mathcal{R}_{1,2}(\boldsymbol{\theta}) = \textstyle\sum_{i,j=1}^d \sqrt{\sum_{m=1}^M \left(w_{i,j}^{(m)}\right)^2} \tag{2.6}$$

is commonly used to enforce sparsity in the excitation functions [130].

Small data amplifies the danger of overfitting; hence the choice of regularizers and their hyperparameters becomes essential. Nevertheless, to control the influence of the penalty in (2.3), all state-of-the-art methods are limited by the use of a single shared hyperparameter α. Ideally, we would have a different hyperparameter to independently control the

effect of the penalty on each parameter of the model. However, the number of parameters, *i.e.*, (d^2M+d), grows quadratically with the dimension of the problem d. To make matters worse, the most common approaches used to fine-tune the choice of hyperparameters, *i.e.*, grid search and random search, become computationally prohibitive when the number of hyperparameters becomes large. Indeed, the search space exponentially increases with the number of hyperparameters. Another approach is to use Bayesian optimization of hyperparameters, but the cost of doing this also becomes prohibitive as the number of samples required to learn the landscape of its cost function exponentially increases with the number of hyperparameters [114]. We describe the details of our proposed approach in the next section.

2.4 Proposed Learning Approach

We now introduce a novel approach for learning the excitation matrix of a Hawkes process. The approach enables us to use a different hyperparameter for each model parameter and efficiently tune them all by taking into account parameter uncertainty. It is based on the variational expectation-maximization (EM) algorithm and jointly optimizes the model parameters $\boldsymbol{\theta}$, as well as the hyperparameters $\boldsymbol{\alpha}$.

First, we can view regularized MLE as a maximum a posteriori (MAP) estimator of the model where parameters are considered as latent variables. Under this interpretation, regularizers on the model parameters correspond to unnormalized priors on the latent variables. The optimization problem becomes

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\geq 0} \log p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \mathcal{S}) = \arg\max_{\boldsymbol{\theta}\geq 0} \log p(\mathcal{S}|\boldsymbol{\theta}) + \log p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}). \tag{2.7}$$

Therefore, having a better regularizer means having a better prior. In the presence of a long sequence of observations, we want the prior to be as uninformative as possible –*i.e.*, a smaller regularization– as we have access to enough information for the MLE to accurately estimate the parameters of the model. But in the case where we only observe short sequences, we want to use more informative priors –*i.e.*, a larger regularization– to avoid overfitting.

Unfortunately, the MAP estimator cannot adjust the influence of the prior by optimizing over $\boldsymbol{\alpha}$. Indeed, the cost function in (2.7) is unbounded from above[2] and solving Equation (2.7) with respect to $\boldsymbol{\alpha}$ leads to a divergent solution $\frac{1}{\boldsymbol{\alpha}} \to \infty$. To address this issue, we can take a Bayesian approach, integrate out parameters and optimize the evidence (or marginal likelihood) $p_{\boldsymbol{\alpha}}(\mathcal{S})$ instead of the log-likelihood. Such an approach changes

[2]We provide more details on this limitation in Appendix A.1.1.

the optimization problem of Equation (2.7) into

$$\hat{\boldsymbol{\alpha}} = \arg\max_{\boldsymbol{\alpha} \geq 0} \int p(\mathcal{S}|\boldsymbol{\theta}) p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}) d\boldsymbol{\theta} = \arg\max_{\boldsymbol{\alpha} \geq 0} \; p_{\boldsymbol{\alpha}}(\mathcal{S}). \tag{2.8}$$

Unlike the MAP objective function, maximizing the evidence over $\boldsymbol{\alpha}$ does not lead to a degenerate solution because it is upper bounded by the likelihood. However, this optimization problem can be solved only for simple models where the integral has a closed form, which requires a conjugate prior to the likelihood. Therefore, we use variational inference to estimate the evidence and develop a variational EM algorithm to optimize our objective with respect to $\boldsymbol{\alpha}$.

2.4.1 Variational Expectation-Maximization Algorithm

Variational inference

The derivation of the variational objective is as follows. First postulate a variational distribution $q_{\boldsymbol{\gamma}}(\boldsymbol{\theta})$, parameterized by the variational parameters $\boldsymbol{\gamma}$, approximating the posterior $p(\boldsymbol{\theta}|\mathcal{S})$. The variational parameters $\boldsymbol{\gamma}$ are chosen such that the Kullback–Leibler divergence between the true posterior $p(\boldsymbol{\theta}|\mathcal{S})$ and the variational distribution $q_{\boldsymbol{\gamma}}(\boldsymbol{\theta})$ is minimized. It is known that minimizing $\mathrm{KL}\left[q_{\boldsymbol{\gamma}}(\boldsymbol{\theta}) \| p(\boldsymbol{\theta}|\mathcal{S})\right]$ is equivalent to maximizing the *evidence lower-bound* (ELBO) [19, 138] defined as

$$\mathsf{ELBO}(q_{\boldsymbol{\gamma}}, \boldsymbol{\alpha}) \coloneqq \mathbb{E}_{q_{\boldsymbol{\gamma}}}\left[\log p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \mathcal{S})\right] - \mathbb{E}_{q_{\boldsymbol{\gamma}}}[\log q_{\boldsymbol{\gamma}}(\boldsymbol{\theta})]. \tag{2.9}$$

By invoking Jensen's inequality on the integral

$$p_{\boldsymbol{\alpha}}(\mathcal{S}) = \int p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \mathcal{S}) d\boldsymbol{\theta},$$

we obtain the desired lower bound on the evidence $p_{\boldsymbol{\alpha}}(\mathcal{S}) \geq \mathsf{ELBO}(q_{\boldsymbol{\gamma}}, \boldsymbol{\alpha})$ where, by maximizing $\mathsf{ELBO}(q_{\boldsymbol{\gamma}}, \boldsymbol{\alpha})$ with respect to $\boldsymbol{\gamma}$, the bound becomes tighter.

For simplicity, we adopt the mean-field assumption by choosing a variational distribution $q_{\boldsymbol{\gamma}}(\boldsymbol{\theta})$ that factorizes over the latent variables[3]. As the parameters $\boldsymbol{\theta}$ are non-negative, a good candidate to approximate the posterior is a log-normal distribution. We define the variational parameters $\boldsymbol{\gamma} = \{\boldsymbol{\nu}, e^{\boldsymbol{\sigma}}\}$ as the mean and the standard deviation of $q_{\boldsymbol{\gamma}}$. We denote the standard deviation by $e^{\boldsymbol{\sigma}}$ because we optimize its log to naturally ensure its positivity and the stability of the optimization procedure. Although we present our learning approach for the log-normal distribution, it is easily generalizable to other distributions.

[3]This assumption can be relaxed using more advanced techniques, such as importance weighted autoencoders (IWAE) [31], at the cost of having a higher computational complexity. However, our experiments have not shown significant performance gain with IWAE.

Variational EM algorithm

In order to efficiently optimize the ELBO with respect to both the variational parameters $\boldsymbol{\gamma}$ and the hyperparameters $\boldsymbol{\alpha}$, we use the variational EM algorithm that iterates over the two following steps: The E-step maximizes the ELBO with respect to the variational parameters $\boldsymbol{\gamma}$ in order to get a tighter lower-bound on the evidence; and the M-step updates the hyperparameters $\boldsymbol{\alpha}$ with a closed form update. Details of the two steps are as follows.

E-step. The E-step maximizes the ELBO with respect to the variational parameters $\boldsymbol{\gamma}$ to make the variational distribution $q_{\boldsymbol{\gamma}}(\boldsymbol{\theta})$ close to the exact posterior $p(\boldsymbol{\theta}|\mathcal{S})$ and to ensure that the ELBO is a good proxy for the evidence. To evaluate the ELBO, we use the black-box variational-inference optimization [66, 104]. We re-parameterize the model as

$$\boldsymbol{\theta} = g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}) = \exp(\boldsymbol{\nu} + e^{\boldsymbol{\sigma}} \odot \boldsymbol{\varepsilon}),$$

where $\boldsymbol{\varepsilon}$ is a $d^2M + d$ vector, with each element following a normal distribution $\mathcal{N}(0, 1)$. $\odot$ denotes the element-wise product. This trick enables us to rewrite the first intractable expectation term of the ELBO in (2.9) as

$$\mathbb{E}_{q_{\boldsymbol{\gamma}}} \left[\log p_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \mathcal{S})\right] = \mathbb{E}_{\boldsymbol{\varepsilon}\sim\mathcal{N}(0,I)} \left[\log p_{\boldsymbol{\alpha}} \left(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}), \mathcal{S}\right)\right]. \tag{2.10}$$

The second term of the ELBO in (2.9) is the entropy of the posterior. For the log-normal distribution, it can be expressed, up to a constant, as $\sum_{\nu_i,\sigma_i}(\nu_i + \sigma_i)$. Hence, the ELBO can be estimated by Monte-Carlo integration as

$$\mathsf{ELBO}(\boldsymbol{\gamma}, \boldsymbol{\alpha}) \approx \frac{1}{L} \sum_{\ell=1}^{L} \log p_{\boldsymbol{\alpha}} \left(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S}\right) + \sum_{\nu_i+\sigma_i} (\nu_i + \sigma_i), \tag{2.11}$$

where L is the number of Monte-Carlo samples $\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_L$. Note that the first term of (2.11) is the cost function for the MAP problem (2.7) evaluated at $\boldsymbol{\theta} = g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell)$ for $\ell \in [L]$. Hence, the E-step summarizes into maximizing the right-hand side of (2.11) with respect to $\boldsymbol{\gamma}$ using gradient descent.

M-step. In the M-step, the ELBO is used as a proxy for the evidence $p_{\boldsymbol{\alpha}}(\mathcal{S})$ and is maximized with respect to the hyperparameters $\boldsymbol{\alpha}$. Again, we rely on the re-parameterization technique and compute the unbiased estimate of the ELBO in (2.11). The maximum of the estimate (2.11) with respect to $\boldsymbol{\alpha}$ has a closed form that depends on the choice of prior. Indeed, by rewriting the joint distribution $\log p_{\boldsymbol{\alpha}}(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S})$ as

$$\log p_{\boldsymbol{\alpha}}(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S}) = \log p(\mathcal{S}|g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell)) + \log p_{\boldsymbol{\alpha}}(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell)), \tag{2.12}$$

the first term –the likelihood– is not a function of $\boldsymbol{\alpha}$ and only the second term –the prior– is a function of $\boldsymbol{\alpha}$. Hence, maximizing the joint distribution

$$\sum_{\ell=1}^{L} \log p_{\boldsymbol{\alpha}} \left(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S} \right)$$

over $\boldsymbol{\alpha}$ amounts to maximizing the prior

$$\sum_{\ell=1}^{L} \log p_{\boldsymbol{\alpha}} \left(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell) \right).$$

To avoid fast changes in $\boldsymbol{\alpha}$ due to the variance of the Monte-Carlo integration, we take an update similar to the one by Bamler et al. [14] and take a weighted average between the current estimate and the maximizer of the current Monte-Carlo estimate of the ELBO as

$$\boldsymbol{\alpha} \leftarrow \zeta \cdot \boldsymbol{\alpha} + (1 - \zeta) \cdot \arg\max_{\tilde{\boldsymbol{\alpha}}} \frac{1}{L} \sum_{\ell=1}^{L} \log p_{\tilde{\boldsymbol{\alpha}}} \left(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S} \right), \tag{2.13}$$

where $\zeta \in [0, 1]$ is the momentum term. Algorithm 2.1 summarizes the proposed variational EM approach. The computational complexity of the inner-most loop of Algorithm 2.1 is L times the complexity of an iteration of gradient descent on the log-likelihood. However, as observed by recent studies in variational inference, using $L = 1$ is usually sufficient in many applications [66]. Hence, we use $L = 1$ in all our experiments, leading to the same computational complexity per-iteration as MLE using gradient descent.

Algorithm 2.1 Variational EM algorithm for Multivariate Hawkes Processes

Input: Sequence of observations $\mathcal{S} = \{(t_n, i_n)\}_{n=1}^{N}$. Initial values for $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Momentum term $0 \leq \zeta < 1$. Sample size L of Monte-Carlo integrations. Number of iterations T_E and T_{EM} of E-steps and EM-steps. Learning rate η.

for $t \leftarrow 1, \ldots, T_{EM}$ **do**
  for $t \leftarrow 1, \ldots, T_E$ **do** ▷ E step
    Sample Gaussian noise $\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_L \sim \mathcal{N}(0, I)$.
    Evaluate the ELBO using Equation (2.11).
    Update $\boldsymbol{\nu} \leftarrow \boldsymbol{\nu} + \eta(\nabla_{\boldsymbol{\nu}} f(\boldsymbol{\nu}, \boldsymbol{\sigma}, \boldsymbol{\varepsilon}; \boldsymbol{\alpha}) + \mathbf{1})$.
    Update $\boldsymbol{\sigma} \leftarrow \boldsymbol{\sigma} + \eta(\nabla_{\boldsymbol{\sigma}} f(\boldsymbol{\nu}, \boldsymbol{\sigma}, \boldsymbol{\varepsilon}; \boldsymbol{\alpha}) + \mathbf{1})$.
  end for
  Sample L Gaussian noise $\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_L$. ▷ M step
  Update $\boldsymbol{\alpha}$ using Equation (2.13).
end for

Output: $\boldsymbol{\alpha}$, $\boldsymbol{\gamma}$

Choice of Prior

In this section, we provide the probabilistic interpretation as a prior of several commonly used regularizers.

L_2-regularizer. The most commonly used regularizer is certainly the L_2-regularizer $w^2/(2\alpha)$ discussed in (2.5). This regularizer can be interpreted as a zero-mean Gaussian distribution over the weights taking the form

$$p_\alpha(w) = \frac{1}{\sqrt{2\pi\alpha}} \exp\left(-\frac{w^2}{2\alpha}\right), \tag{2.14}$$

where α is the variance. Hence, setting the derivative of the log of the distribution to zero, we find that the closed-form update for the M-Step is

$$\arg\min_{\tilde{\alpha}} \frac{1}{L}\sum_{\ell=1}^{L} \log p_{\tilde{\alpha}}(g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell), \mathcal{S}) = \frac{1}{L}\sum_{\ell=1}^{L} g_{\boldsymbol{\gamma}}(\boldsymbol{\varepsilon}_\ell)^2. \tag{2.15}$$

The update rules for the other priors can be found similarly.

L_1-regularizer. This regularizer, also known as *lasso* regularizer, is often considered as a convex surrogate for the L_0 (pseudo) norm to promote sparsity in the parameters [141]. It can be interpreted as a Laplace distribution over the weights, *i.e.*,

$$p_\alpha(w) = \frac{1}{2\alpha} \exp\left(-\frac{|w|}{\alpha}\right). \tag{2.16}$$

Low-rank regularizer. To achieve a low-rank integrated excitation matrix $\boldsymbol{\Phi} \coloneqq [\int_{\mathbb{R}} \phi_{i,j}(t)dt]$, a nuclear norm penalty on $\boldsymbol{W} \coloneqq [w_{i,j}]$ is used as a regularizer by Zhou et al. [141] to enable clustering structures in $\boldsymbol{W}$ for the parametric case where $M = 1$. In this case, with $\boldsymbol{w}_{\cdot,j} = [w_{1,j} \ldots, w_{d,j}]$, the different $\{\boldsymbol{w}_{\cdot,j}\}_j$ are independent for different j and the prior over $\boldsymbol{w}_{\cdot,j}$ can be expressed as

$$p_\alpha(\boldsymbol{w}_{\cdot,j}) = c \cdot \frac{1}{\alpha^d} \exp\left(-\frac{\|\boldsymbol{w}_{\cdot,j}\|_2}{\alpha}\right), \tag{2.17}$$

where $c > 0$ is a normalizing constant.

Group-lasso regularizer. This regularizer is used by Xu et al. [130] in the non-parametric setting defined in Section 2.3 where the excitation function is approximated by a linear combination of M basis functions, parameterized by $\boldsymbol{w}_{i,j} = \left[w_{i,j}^{(1)}, \ldots, w_{i,j}^{(M)}\right]$.

In this case, the L_2-norm of $\boldsymbol{w}_{i,j}$ is assumed to have a Laplace distribution, *i.e.*,

$$p_\alpha(\boldsymbol{w}_{i,j}) = c \cdot \frac{1}{\alpha^M} \exp(-\frac{\|\boldsymbol{w}_{i,j}\|_2}{\alpha}). \tag{2.18}$$

where $c > 0$ is a normalizing constant.

2.5 Experimental Results

We carry out two sets of experiments to evaluate the performance of our approach compared to the state of the art. First, we perform a link-prediction task on synthetic data to show that our approach can accurately recover the support of the excitation matrix of the process under short sequences. Second, we perform an event-prediction task on real datasets of short sequences to show that our approach outperforms state-of-the-art methods in terms of predictive log-likelihood.

We run our experiments in two different settings. First, in a *parametric* setting where the exponential form of the excitation function is known, we compare our approach (VI-EXP) to the state-of-the-art MLE-based method (MLE-ADM4) from Zhou et al. [141]. Second, we use a *non-parametric* setting where no assumption is made on the shape of the excitation function. We then set the excitation function as a mixture of $M = 10$ Gaussian kernels defined as

$$\kappa_m(t) = (2\pi b^2)^{-1} \exp\left(-\frac{(t-\tau_m)^2}{2b^2}\right), \ \forall m = 1, \dots, M, \tag{2.19}$$

where τ_m and b are the known location and scale of the kernel. In this setting, we compare our approach (VI-SG) to the state-of-the-art MLE-based methods (MLE-SGLP) of Xu et al. [130] with the same $\{\kappa_m(t)\}$[4]. Let us stress that the parametric methods have a strong advantage over the non-parametric ones because they are given the true value of the exponential decay β.

As our VI approach returns a posterior on the parameters, rather than a point estimate, we use the mode of the approximate log-normal posterior as the inferred edges $\{\widehat{w}_{i,j}\}$. For the non-parametric setting, we use $\widehat{w}_{i,j} = \sum_{m=1}^{M} \widehat{w}_{i,j}^{(m)}$. To mimic the regularization schemes of the baselines, we use a Laplacian prior for the edge weights $\{w_{i,j}\}$ to enforce sparsity, and we use a Gaussian prior for the baselines $\{\mu_i\}$. We tune the hyperparameters of the baselines using grid search[5].

[4] We also performed the experiments with other approaches designed for large-scale datasets, but their performance was below that of the reported baselines [2, 76, 77].

[5] More details are provided in Appendix C.1.

2.5.1 Synthetic Data

First, we evaluate the performance of our VI approach on simulated data. We generate random Erdős–Rényi graphs with $d = 50$ nodes and edge probability $p = \log(d)/d$. Then, a sequence of observations is generated from a Hawkes process with exponential excitation functions defined in (2.1) with exponential decay $\beta = 1$. The baselines $\{\mu_i^*\}$ are sampled independently in Unif$[0, 0.02]$, and the edge weights $\{w_{i,j}^*\}$ are sampled independently in Unif$[0.1, 0.2]$. Results are averaged over 30 graphs with 10 simulations each. For reproducibility, a detailed description of the experimental setup is provided in Appendix C.1.

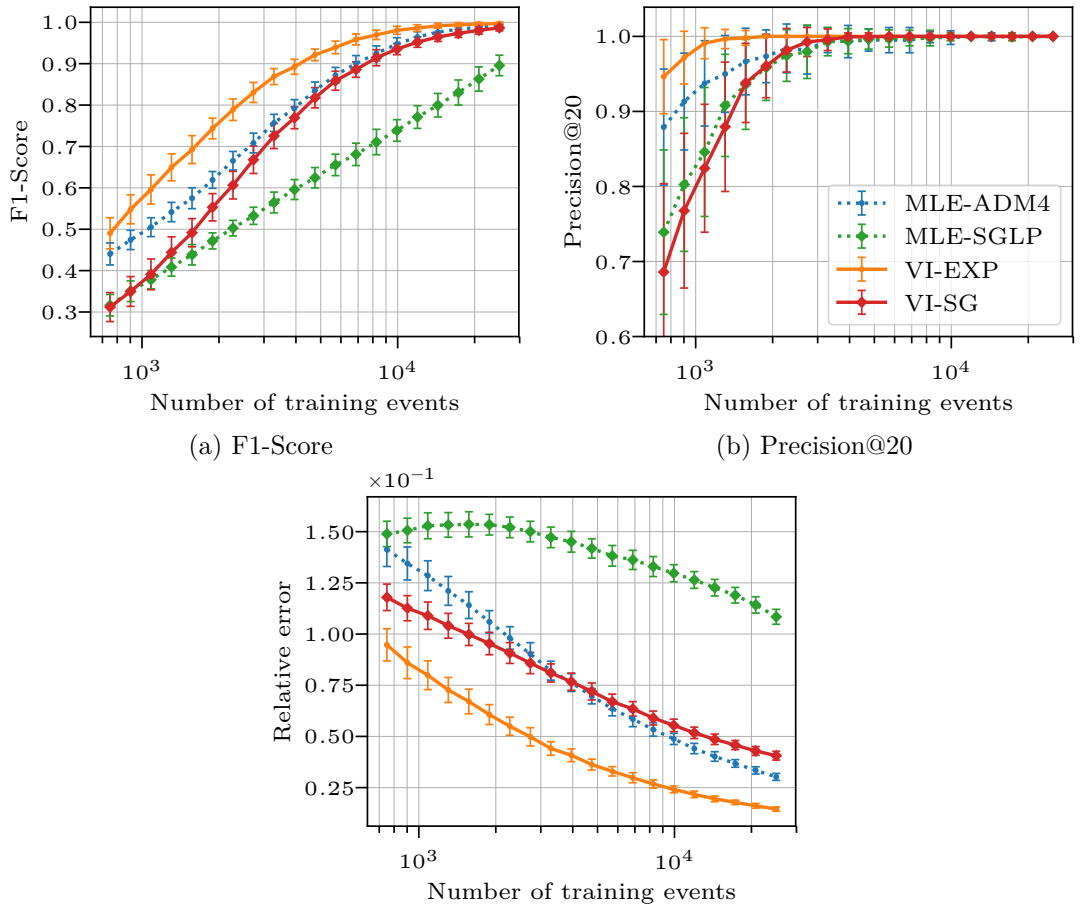
To investigate if the support of the excitation matrix can be accurately recovered under small data, we evaluate the performance of each approach on three metrics [142, 130, 49].

- **F1-score.** We zero-out small weights using a threshold $\eta = 0.04$ and measure the F1-score of the resulting binary edge classification problem. Additional results with varying thresholds η are provided in Appendix B.1.

- **Precision@k.** Instead of thresholding, we also report the precision@k defined by the average fraction of correctly identified edges in the top k largest estimated weights. Since the proposed VI approach gives an estimate of uncertainty via the variance of the posterior, we select the edges with high weights $\widehat{w}_{i,j}$ and low uncertainty, *i.e.*, the edges with ratio of the lowest standard deviation over weight $\widehat{w}_{i,j}$.

- **Relative error.** To evaluate the distance of the estimated weights to the ground truth ones, we use the averaged relative error defined as $|\widehat{w}_{i,j} - w_{i,j}^*|/w_{i,j}^*$ when $w_{i,j}^* \neq 0$, and $\widehat{w}_{i,j}/(\min_{w_{k,l}^*>0} w_{k,l}^*)$ when $w_{i,j}^* = 0$. This metric is more sensitive to errors in small weights $w_{i,j}^*$, and therefore penalizes false positive over false negative errors.

We first investigate the sensitivity of each approach to the amount of data available for training by varying the size of the training set from $N = 750$ to $N = 25\,000$ events, *i.e.*, 15 to 500 events per node. Results are shown in Figure 2.1. Our approach improves the results in both parametric and non-parametric settings for all metrics. The improvements are more substantial in the non-parametric setting. If the accuracy of the top edges is similar for both VI-SG and MLE-SGLP in terms of precision@20, VI-SG improves the F1-score by about 20% with $N = 5\,000$ training events. The reason for this improvement is that MLE-SGLP has a much higher false positive rate, which is hurting the F1-score but does not affect the precision@20. VI-SG is also able to reach the same F1-score as the parametric baseline MLE-ADM4 with only $N = 4\,000$ training events. Note that VI-SG is optimizing $d^2M + d = 25\,050$ hyperparameters with minimal additional cost.



(a) F1-Score (b) Precision@20

(c) Relative Error

Figure 2.1 – Performance with respect to the number of training samples measured by (a) F1-Score, (b) Precision@20, and (c) Relative error. Our VI approaches are shown in solid lines. The non-parametric methods are highlighted with square markers. Results are averaged over 30 random graphs with 10 simulations each (± standard deviation).

In the next experiment, we focus on the non-parametric setting. We fix the length of observation to $N = 5000$ and study the effect of increasing M on the performance of the algorithms. The results are shown in Figure 2.2. We see that our approach is more robust to the choice of M than MLE-SGLP. A possible explanation for this behavior is that MLE-SGLP overfits because of the increasing number of model parameters.

We also investigate the parameters of the model learned by our VI-EXP approach. In Figure 2.3a, we use the variance of the approximated posterior q_γ as a measure of confidence for edge identification, and we report the distribution of ratio of standard deviation over weight $\widehat{w}_{i,j}$ for both the true and false positive edges. Similar results hold between the true and false negative edges. The false positive edges have a higher uncertainty than the true positive ones. This is relevant when we cannot identify all

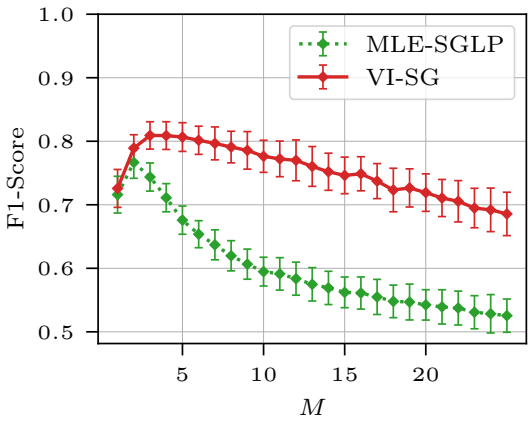


Figure 2.2 – Analysis of the robustness of non-parametric approaches to the number of bases M of excitation functions (for fixed $N = 2\,000$).

edges due to lack of data, even though we still wish to identify a subset of edges with high confidence. Figure 2.3b confirms that, as expected, the optimized weight priors $\boldsymbol{\alpha}$ are much larger for true edges in the ground-truth excitation matrix than for non-edges.

Finally, to evaluate the scalability of our approach, we fixed the number of training events per dimension and analyzed the empirical running time[6] on increasingly large-dimensional problems for both VI-EXP and MLE-ADM4. As shown in Figure 2.4, the empirical running time per iteration of our approach VI-EXP (implemented in Python) scales better than the one of MLE-ADM4 (implemented in C++). Although our gradient descent algorithm requires more iterations to converge, we show in Figure 2.5 that VI-EXP reaches the same F1-score as MLE-ADM4 faster.

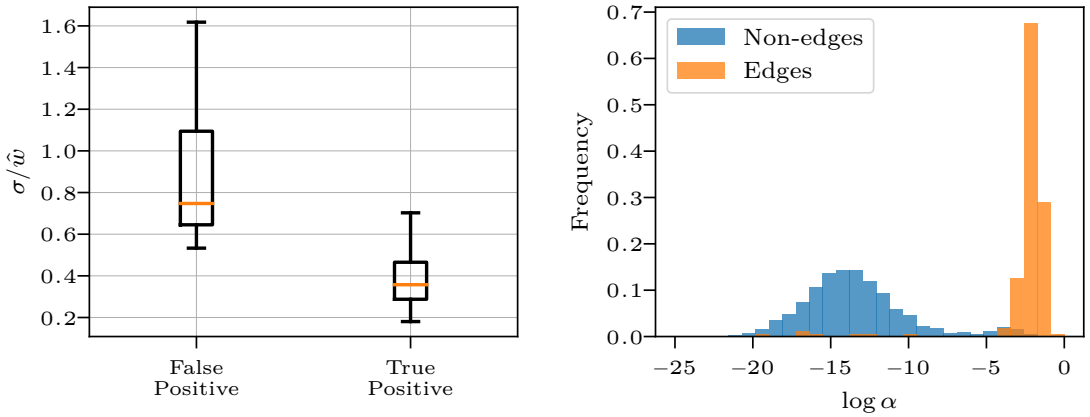
2.5.2 Real Data

We also evaluate the performance of our approach on the following three small datasets:

1. **Epidemics.** This dataset contains records of infection of individuals, along with their corresponding district of residence, during the last Ebola epidemic in West Africa in 2014-2015 [51]. To learn the propagation network of the epidemics, we consider the 54 districts as processes and define infection records as events.

2. **Stock market.** This dataset contains the stock prices of 12 high-tech companies sampled every 2 minutes on the New York Stock Exchange for 20 days in April 2008 [44]. We consider each stock as a process and record an event every time a stock price changes by 0.15% from its current value.

[6] All experiments were run single-threaded on the same machine with a CPU Intel Xeon E5-2680 v3 (Haswell), 2.5 GHz, 30 MB cache.



(a) (b)

Figure 2.3 – Analysis of the uncertainty of the parameters learned by VI-EXP (for fixed $N = 5\,000$). (a) Uncertainty of the inferred edges and (b) histogram of learned $\boldsymbol{\alpha}$. The learned $\boldsymbol{\alpha}$ are smaller for non-edges, and false positive edges have higher uncertainty than the true positive ones.

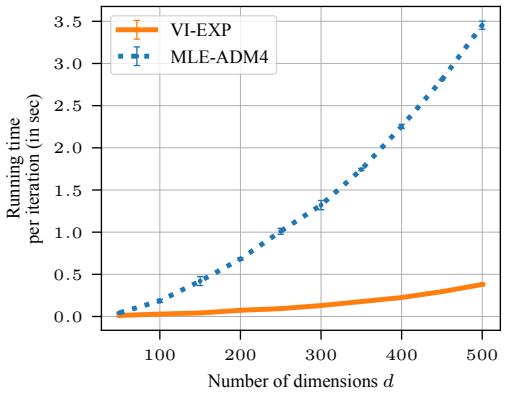


Figure 2.4 – Comparison of running time per-iteration for VI-EXP and MLE-ADM4.

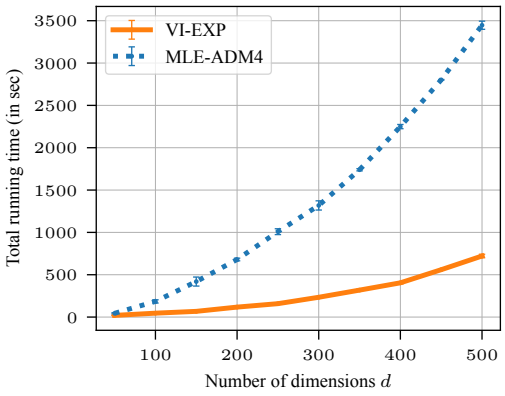


Figure 2.5 – Running time required for our approach VI-EXP to reach the same F1-Score as MLE-ADM4.

3. **Enron email.** This dataset contains emails between employees of Enron from the Enron corpus. We consider all employees with more than 10 received emails as processes and record an event every time an employee receives an email.

We perform an event-prediction task to show that our approach outperforms the state-of-the-art methods in terms of predictive log-likelihood. To do so, we use the first 70% events as training set, and we compute the held-out averaged log-likelihood on the remaining 30%. We present the results in Table 2.1.

We first see that the non-parametric methods outperform the parametric ones on both the Epidemic dataset and the Stock market dataset. This suggests that the exponential excitation function might be too restrictive to fit their excitation patterns. In addition, our non-parametric approach VI-SG significantly outperforms MLE-SGLP on all datasets. The improvement is particularly clear for the Epidemic dataset, which has the smallest number of events per dimension. Indeed, the top edges learned by VI-SG correspond to contiguous districts as expected. This is not the case for MLE-SGLP, for which the top learned edges correspond to districts that are far from each other.

Table 2.1 – Predictive log-likelihood for the models learned on various real datasets.

Dataset	Statistics		Averaged predictive log-likelihood			
	#dim (d)	#events ($\lvert\mathcal{S}\rvert$)	**VI-SG**	MLE-SGLP	**VI-EXP**	MLE-ADM4
Epidemics	54	5 349	$\mathbf{-2{,}06}$	$-3{,}03$	$-4{,}31$	$-4{,}61$
Stock market	12	7 089	$\mathbf{-1{,}00}$	$-2{,}45$	$-2{,}82$	$-2{,}81$
Enron email	143	74 294	$-0{,}42$	$-1{,}01$	$\mathbf{-0{,}23}$	$-0{,}40$

2.6 Summary

In this chapter, we proposed a novel approach to learn the excitation matrix of a multivariate Hawkes process in the presence of short observation sequences. We observed that state-of-the-art methods are sensitive to the amount of data used for training and showed that the proposed approach outperforms these methods when only short training sequences are available. The common tool to tackle this problem is to design smarter regularization schemes. However, all maximum likelihood-based methods suffer from a common problem: all the model parameters are regularized equally with a few hyperparameters. We developed a variational expectation maximization algorithm that is able to (1) optimize over an extended set of hyperparameters, with almost no additional cost and (2) take into account the uncertainty of the learned model parameters by fitting a posterior distribution over them. We performed experiments on both synthetic and real datasets and showed that our approach outperforms state-of-the-art methods under small-data regimes.

3 Learning Hawkes Processes under Synchronization Noise

In this chapter[1], we address the problem of learning the causal structure of the Hawkes process when the timestamps of events cannot be recorded accurately. In particular, we introduce the so-called *synchronization noise*, where the stream of events generated by each dimension is subject to a random and unknown time shift. We characterize the sensitivity of the classic maximum likelihood estimator to synchronization noise and highlight the challenges posed by such temporal noise. We introduce a new approach for learning the causal structure in the presence of noise. Our experimental results show that our approach accurately recovers the causal structure of Hawkes processes for a wide range of noise levels, and significantly outperforms classic estimation methods.

3.1 Introduction

Learning the excitation matrix of a Hawkes process, which encodes the Granger causal structure between the processes from a set of observations, has been the focus of recent work[2]. All these studies assumes that the observations are noiseless, that is, the arrival times of the events are recorded accurately without any delay. To the best of our knowledge, no work to date has considered learning the causal structure of a noisy Hawkes process. Recent studies tackled the inference of Hawkes processes with missing data [111, 131], but did not consider noisy (delayed) observations. The inference of temporal point processes in the presence of noisy observations has been studied for non-parametric estimators of spatial Poisson processes [15, 32]. However, these studies mostly focus on the special case of independent and known noise and cannot be applied to Hawkes processes.

We study the problem of learning Hawkes processes in the presence of observation noise. More precisely, we consider *synchronization noise*, where the stream of events generated by each source –or dimension– is subject to a random and unknown time shift. This model

[1] This chapter is based on Trouleau et al. [116].

[2] We refer the reader to Section 1.4 for a detailed discussion.

captures situations where no perfect clock time synchronization is available at different sources, or when the observation process itself introduces source-dependent delays. As an example of the former, consider a network of sensors that record events such as neural spikes or earthquake shocks. It is often the case that the sensors are not perfectly synchronized, because they each rely on a local clock to time-stamp events. As an example of the latter, consider processes where an event can only be observed indirectly after a delay, such as through the symptoms of an infectious disease that manifest themselves some time after the actual infection. We will show that synchronization noise can severely harm the estimation performance of state-of-the-art learning methods.

Our contribution is two-fold. First, we demonstrate the vulnerability of the state-of-the-art learning algorithms to noisy observations. Second, we provide a novel estimation approach for learning the causal structure of a Hawkes process in the presence of synchronization noise. Unlike previous works on the inference of point processes with noise [15, 32], our approach does not assume that the noise is sampled from a known distribution. Our approach is based on the maximum-likelihood estimation of a novel model called desynchronized multivariate Hawkes process (DESYNC-MHP) in which the parameters of interest consist of the Hawkes process parameters along with the noise. In other words, given a set of observed data, our approach learns the Hawkes process with synchronization noise that maximizes the log-likelihood with respect to both the noise and the parameters of the process. Such log-likelihood function is smooth with respect to the Hawkes process parameters, yet non-convex and non-smooth with respect to the noise parameters. We show that maximizing a smoothed version of this objective function with respect to both the noise and the Hawkes process parameters recovers the excitation matrix and hence the causal structure of the process.

Outline of the Chapter. The chapter is organized as follows. In Section 3.2, we provide some preliminary definitions and notations. We introduce the synchronization noise framework in Section 3.3 and show how it biases the classic maximum likelihood estimation algorithm that assumes the observations to be noiseless. In Section 3.4, we introduce our methodology to learn Hawkes processes under synchronization noise. Finally, we demonstrate the performance of our approach on synthetic simulations, and we validate it on a dataset of neuronal spike trains in Section 3.5.

3.2 Preliminary Definitions

Prior to discussing our results, we introduce the basic notations and definitions used in this chapter. Detailed notations will be introduced along the way. Recall that, as defined in Section 1.3, a d-dimensional Hawkes process is a collection of d univariate temporal point processes $N_i(t)$, $i = 1, \ldots, d$, also called dimension, with a particular form of the conditional intensity function introduced in (1.18). The dynamics of influence between

the processes are captured by the excitation matrix $\mathbf{\Phi}(t) \coloneqq [\phi_{i,j}(t)]$. A common choice for the excitation function $\phi_{i,j}(t)$ is an exponential kernel of the form

$$\phi_{i,j}(t) \coloneqq w_{i,j} e^{-\beta t} \mathbb{1}_{\{t>0\}}, \tag{3.1}$$

where $w_{i,j}$ captures the strength of influence and β captures the time constant [47, 100, 111, 132, 141]. In this chapter, we present our learning approach for the exponential kernel in (3.1), but it is applicable to more general forms of parametric kernels.

Suppose that, during a time period $[t_0, T]$, we observe a sequence of events

$$\mathcal{S} = \{(i_n, t_n)\}_{n \geq 1},$$

where $t_n \in [t_0, T]$ is the timestamp of the n-th event and $i_n \in [d]$ is its dimension. Let $\boldsymbol{\theta}$ denote the parameters of the Hawkes process, which consist of the excitation matrix $\{w_{i,j}\}$ and the background intensities $\{\mu_i\}$. Maximum likelihood estimation can be used to learn $\boldsymbol{\theta}$ from the observations $\mathcal{S}$. Similar to the definition in Chapter 2, the log-likelihood of $\mathcal{S}$ given $\boldsymbol{\theta}$ for the Hawkes process is given by

$$\log p(\mathcal{S}|\boldsymbol{\theta}) = \sum_{(i_n,t_n)\in\mathcal{S}} \log \lambda_{i_n}(t_n|\mathcal{H}_{t_n}) - \sum_{i=1}^{d} \int_{t_0}^{T} \lambda_i(t|\mathcal{H}_t)dt. \tag{3.2}$$

It can be shown that (3.2) is convex for exponential kernels if the exponential decay β is known [11]. It is therefore common practice to define β as a hyperparameter and to apply maximum likelihood estimation only to

$$\boldsymbol{\theta} \coloneqq \{\{\mu_i\}_{i=1}^{d}, \{w_{i,j}\}_{i,j=1}^{d}\} \in \mathbb{R}_{\geq 0}^{d(d+1)}.$$

3.3 Noisy Observation Framework

In this section, we introduce a particular form of noise, called synchronization noise. We demonstrate its destructive effect on the classic maximum likelihood (ML) estimation methodology, which assumes noiseless observations.

3.3.1 Synchronization Noise

With synchronization noise, all the arrivals within a dimension are shifted equally by an unknown offset. In other words, for every dimension i, there exists z_i, such that the observed data is

$$\widetilde{\mathcal{S}} \coloneqq \{(i_n, \tilde{t}_n)\}_{n\geq 1} = \{(i_n, t_n + z_{i_n})\}_{n\geq 1}.$$

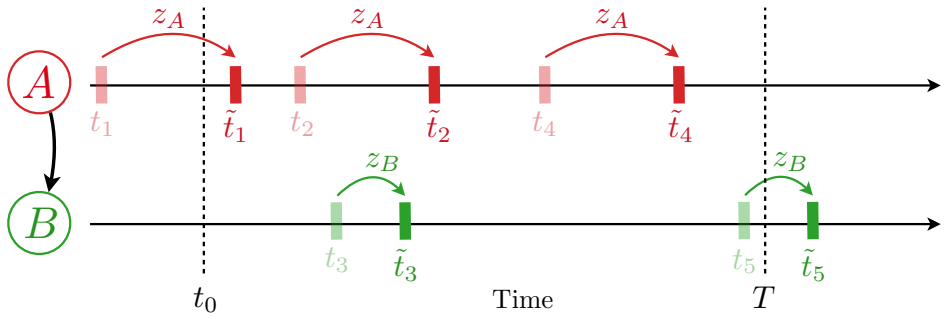


Figure 3.1 – Illustration of the synchronization noise model on a simple two-dimensional Hawkes process, with type A influencing type B. Noisy events are displayed in solid ticks whereas the original events are shown in transparent ticks. The arrows illustrate the time shift introduced by the noise.

In other words, each event t_n is shifted in time by an offset of z_{i_n} shared by all events of type i_n. We denote the collection of noise variables by $\boldsymbol{z} = \{z_i\}_{i=1}^{d}$. Because of boundary effects due to the finite observation window, the number of noisy observations may differ from the number of noiseless events in $[t_0, T]$ as some events can enter or escape the observation window.

To make this more concrete, Figure 3.1 shows a simple example of synchronization noise for a 2-dimensional Hawkes process with types $\{A, B\}$. The synchronization noise values $\{z_A, z_B\}$ do not change the relative orders of the arrivals within a dimension but it affects the relative orders of the arrivals between different dimensions. For instance, the event t_2 of type $i_2 = A$ comes before the event t_3 of type $i_3 = B$, namely, $t_2 < t_3$, but

$$t_2 + z_{i_2} = t_2 + z_A = \tilde{t}_2 > \tilde{t}_3 = t_3 + z_{i_3} = t_3 + z_B,$$

so their order is reversed. Some events can also enter (or escape) the observation window, such as t_1 of type A (or t_5 of type B).

3.3.2 Effect of Noise on Classic Inference Methods

The synchronization noise may swap the relative order of arrivals between different dimensions, which results in estimation errors for classic inference methods, such as ML estimation. Consider once again the simple network of two processes shown in Figure 3.1. In this example, the causal graph contains a single edge $A \to B$, implying that events of types A cause future events of type B (but not the other way around). Figure 3.2 displays the result of ML estimation with synchronization noise for these two processes. When $z_A < z_B$, events of type B tend to occur after their cause (parent) of type A, which leads ML estimation to correctly identify the causal direction $A \to B$. However, as $z_A > z_B$, the causes and effects begin to blur. This forces ML estimation to learn edges in both directions. Finally, as the difference between z_A and z_B gets large, the inferred

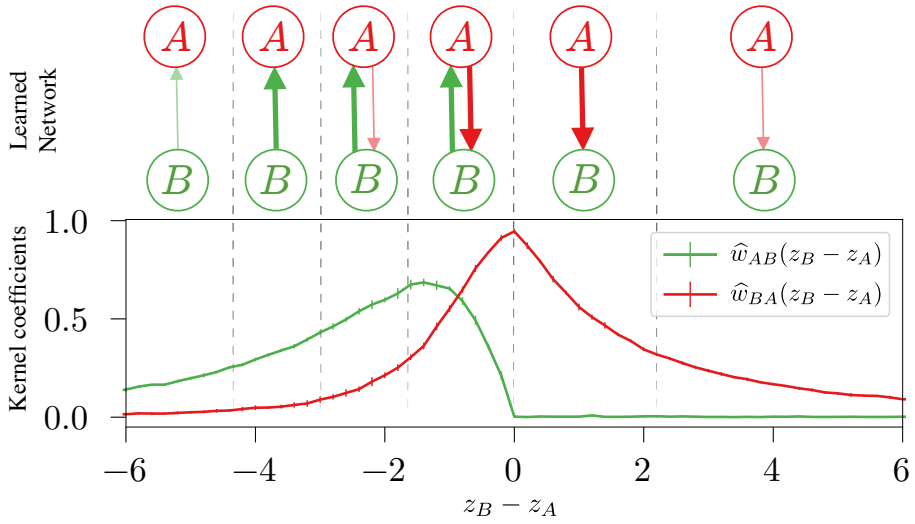


Figure 3.2 – Maximum likelihood estimate on the toy example of Figure 3.1 as a function of noise values. When $z_B - z_A < 0$, maximum-likelihood estimation detects edges in both directions, *i.e.*, $\widehat{w}_{AB}$ and $\widehat{w}_{BA}$ are both positive.

dependency between A and B decreases. This is the reason explaining the convergence of the kernel coefficients to zero.

3.4 Inference under Synchronization Noise

In this section, we introduce a new robust inference approach for learning Hawkes processes in the presence of synchronization noise. We first note that, if the value of the noise $\boldsymbol{z}$ is known, we can simply subtract the value of the noise from each arrival time, and the problem reduces to the inference of a standard (noiseless) Hawkes process. Conditioning on the noise $\boldsymbol{z}$, the log-likelihood in (3.2) can hence be written as the conditional log-likelihood

$$\begin{aligned}\log p(\widetilde{\mathcal{S}}|\boldsymbol{z},\boldsymbol{\theta}) &= \log p\left(\{(i_n,\tilde{t}_n - z_{i_n})\}_{n\geq 1}\middle|\boldsymbol{\theta}\right)\\ &= \sum_{(i_n,\tilde{t}_n)\in\widetilde{\mathcal{S}}} \log\lambda_{i_n}\left(\tilde{t}_n - z_{i_n}|\widetilde{\mathcal{H}}_{\tilde{t}_n - z_{i_n}}\right) - \sum_{i=1}^{d}\int_{t_0-z_i}^{T-z_i}\lambda_i(t|\widetilde{\mathcal{H}}_t)dt, \qquad (3.3)\end{aligned}$$

where $\widetilde{\mathcal{H}}_t = \{(i_n,\tilde{t}_n) \mid \tilde{t}_n < t\}$ is the observed history of the (noisy) processes up to time t. It is important to notice that (3.3) is a function of the observed history $\widetilde{\mathcal{H}}_t$ due to the conditional intensity function terms. Since the synchronization noise can change the order of the arrivals in different dimensions and consequently the value of the conditional intensity function, it can also change the above conditional log-likelihood. Hence, the noise offset $\boldsymbol{z}$ affects the Hawkes process parameters $\boldsymbol{\theta}$ maximizing (3.3).

We define a new multivariate point process called *desynchronized multivariate Hawkes process* (DESYNC-MHP) that is a Hawkes process with synchronization noise. The parameters of this model are $\{\boldsymbol{z}, \boldsymbol{\theta}\}$. In other words, a DESYNC-MHP with parameters $\{\boldsymbol{z}, \boldsymbol{\theta}\}$ is a Hawkes process with parameter $\boldsymbol{\theta}$, where each dimension $i \in [d]$ is affected by the synchronization noise offset z_i. Therefore, the log-likelihood function of this model, given a set of observed arrivals $\widetilde{\mathcal{S}}$, can be written as (3.3). Hence, ML estimation for the DESYNC-MHP amounts to solving the optimization problem

$$\hat{\boldsymbol{z}}, \hat{\boldsymbol{\theta}} = \operatorname*{arg\,max}_{\boldsymbol{z} \in \mathbb{R}, \boldsymbol{\theta} \geq 0} \log p(\widetilde{\mathcal{S}} | \boldsymbol{z}, \boldsymbol{\theta}). \tag{3.4}$$

An alternative approach to directly maximizing the log-likelihood is to consider the noise as a latent variable and to use the EM algorithm. However, such an approach requires to evaluate the posterior distribution, which is intractable because of its coupling with the ordering of the events. It is therefore easier to solve (3.4) directly. This approach still introduces new challenges that we will address next.

Challenges

For a *given* noise variable $\boldsymbol{z}$, maximizing (3.3) with respect to the Hawkes process parameters $\boldsymbol{\theta}$ results in the ML estimation for the noiseless Hawkes process, which can be often solved efficiently. For instance, in the exponential kernel setting, when $\boldsymbol{\theta} = \{\{\mu_i\}_{i=1}^d, \{w_{i,j}\}_{i,j=1}^d\}$, the problem is smooth and convex, and therefore the parameters can be easily estimated using first-order methods [47, 100, 132, 141].

In contrast, the objective function in (3.3) is neither smooth nor continuous with respect to the noise $\boldsymbol{z}$. Recall that in (3.3) the intensity function (1.18) depends on the history $\widetilde{\mathcal{H}}_t$ of the process. However, synchronization noise can invert the order of arrivals in different dimensions, and consequently it can change the past events of some arrivals. This change in the ordering of events creates discontinuities in the likelihood and makes it particularly challenging to optimize.

To observe this concretely, consider a 2-dimensional Hawkes process with only two arrival times t_1 and t_2 ($t_1 < t_2$), in dimensions 1 and 2, respectively. Suppose that the observed arrival times $\tilde{t}_1$ and $\tilde{t}_2$, are such that $\tilde{t}_1 < \tilde{t}_2$. The effect of dimension 1 on dimension 2 is captured by

$$\phi_{2,1}(\tilde{t}_2 - z_2 - \tilde{t}_1 + z_1) = w_{2,1}\, e^{-\beta(\tilde{t}_2 - z_2 - \tilde{t}_1 + z_1)} \mathbb{1}_{\{\tilde{t}_2 - z_2 - \tilde{t}_1 + z_1 > 0\}}$$

Hence, for a given z_1, as z_2 increases, the excitation function increases until $z_2 = \tilde{t}_2 - \tilde{t}_1 + z_1$. At this point, the arrival orders are switched and the effect of the arrival at t_1 on the

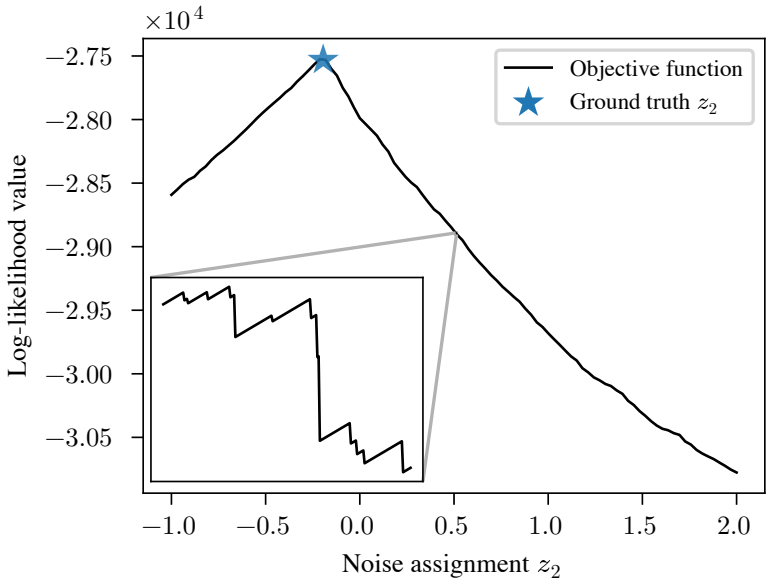


Figure 3.3 – Illustration of the discontinuities of the objective function (3.3) for a 2-dimensional Hawkes process as a function of z_2, when z_1 is fixed to its true value and $\beta = 1$. The inset shows a fine zoom on the objective function in 0.5 ± 0.005.

arrival at t_2 disappears. Formally, at $\tau = \tilde{t}_2 - z_2 - \tilde{t}_1 + z_1$, we have

$$\lim_{\tau \to 0^+} \phi_{2,1}(\tau) = w_{2,1} \quad \neq \quad 0 = \lim_{\tau \to 0^-} \phi_{2,1}(\tau).$$

This results in a discontinuity in the objective function.

Figure 3.3 illustrates the objective function as a function of z_2, when z_1 is fixed to its true value, for a two-dimensional process. These discontinuities in the conditional log-likelihood function will prevent gradient-based algorithms from converging. Even worse, the objective function is particularly ill-conditioned: it decreases at the points of discontinuity, but increases everywhere in between. The presence of synchronization noise therefore transforms the computationally efficient estimation of the Hawkes process parameters into a particularly ill-conditioned optimization problem.

Below, we discuss our approach to tackle this issue in two steps. We first introduce a novel approach for smoothing the objective function, which allows us to subsequently find an optimum solution by using stochastic gradient descent.

Smoothing the objective function. Recall that the source of the discontinuities (jumps) in the objective function are the swapped arrivals and the discontinuities of the excitation kernels at $t = 0$. If the excitation kernels $\{\phi_{i,j}(t)\}$ were differentiable for all $t \in \mathbb{R}$, such sudden jumps in the intensity function would be avoided and consequently

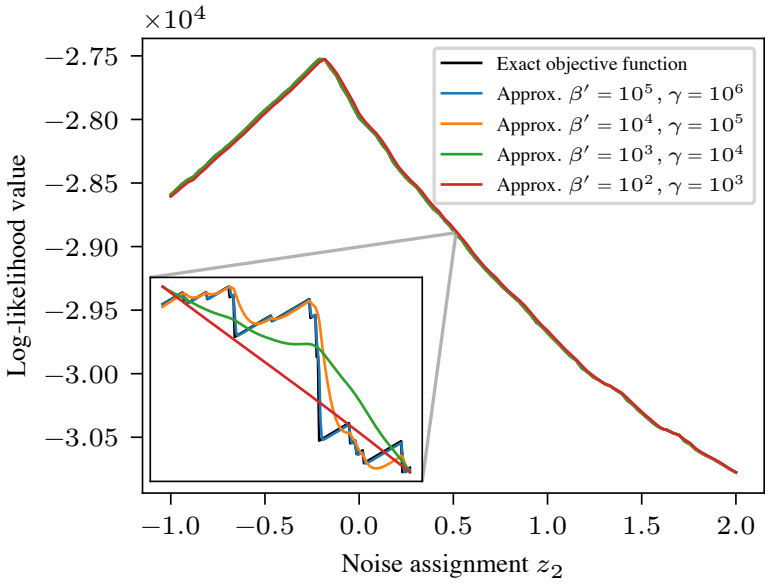


Figure 3.4 – Illustration of the smoothing of the objective function (3.3) for a 2-dimensional Hawkes process as a function of z_2, when $z_1 = z_1^*$ and $\beta = 1$. The inset shows a fine zoom on the objective function in 0.5 ± 0.005.

the likelihood function would be smooth. This observation leads us to approximate the excitation kernels with functions that are differentiable everywhere. For instance, one candidate for approximating the exponential kernel is

$$\tilde{\phi}_{i,j}(t) \triangleq w_{i,j} \left(\sigma(\gamma t) e^{-\beta t} + (1 - \sigma(\gamma t)) e^{\beta' t} \right), \tag{3.5}$$

where $\sigma(t) = 1/(1 + e^{-t})$ is the sigmoid function[3]. Because

$$\lim_{\beta', \gamma \to +\infty} \tilde{\phi}_{i,j}(t) = \phi_{i,j}(t), \tag{3.6}$$

the approximated kernel can be made arbitrarily close to $\phi_{i,j}(t)$. Selecting β' and γ large enough will therefore preserve the causal structure of the Hawkes process. Figure 3.4 illustrates how $\tilde{\phi}_{i,j}(t)$ affects the objective function for various values of β' and γ.

Stochastic gradient descent. The kernel approximation (3.5) addresses the non-smoothness of the objective function with respect to the noise $\boldsymbol{z}$. But the issue of convexity remains, as illustrated in the inset of Figure 3.4 for large values of β'. This means that choosing the right β' is crucial. On the one hand, a small β' makes the objective function smoother and removes some local minima. On the other hand, a small β' degrades the quality of the approximation and hence introduces a larger bias in the optimization problem.

[3]Note that this choice of kernel is non-causal, in the sense that the kernels are non-zero for $t < 0$.

Stochastic gradient descent (SGD) is often used to escape local minima in non-convex optimization. In our case, SGD randomizes the discontinuities, and hence enables us to evade the local minima. We apply a mini-batch version of SGD with a set of C independent observations $\{\widetilde{\mathcal{S}}_1, \ldots, \widetilde{\mathcal{S}}_C\}$. Because of the ergodicity of stationary Hawkes processes, a set of short independent observations of a Hawkes process is statistically equivalent to a single long observation of that Hawkes process.

Algorithm 3.1 summarizes the steps of our approach[4]. Since smoothing is only necessary for optimizing $\log p(\widetilde{\mathcal{S}}|\boldsymbol{z}, \boldsymbol{\theta})$ with respect to $\boldsymbol{z}$, we use the gradient[5] of the smooth approximation of the log-likelihood, denoted by $\nabla_{\boldsymbol{z}} \log \widetilde{p}(\widetilde{\mathcal{S}}|\boldsymbol{z}, \boldsymbol{\theta})$, to update $\boldsymbol{z}$, and we keep the gradient of the exact log-likelihood to update the Hawkes process parameters $\boldsymbol{\theta}$, denoted by $\nabla_{\boldsymbol{\theta}} \log p(\widetilde{\mathcal{S}}_k|\boldsymbol{z}_k, \boldsymbol{\theta}_k)$.

Algorithm 3.1 DESYNC-MHP Maximum Likelihood Estimation

Input: Data $\{\widetilde{\mathcal{S}}_1, \ldots, \widetilde{\mathcal{S}}_C\}$, hyperparameters (β, β', γ).

Initialize $\boldsymbol{z}_0$ and $\boldsymbol{\theta}_0$ to random values
$k \leftarrow 0$
repeat
  $\widetilde{\mathcal{S}}_k \sim \text{Uniform}\{\widetilde{\mathcal{S}}_1, \ldots, \widetilde{\mathcal{S}}_C\}$
  $\boldsymbol{z}_{k+1} \leftarrow \boldsymbol{z}_k + \delta_k \, \nabla_{\boldsymbol{z}} \log \widetilde{p}(\widetilde{\mathcal{S}}_k|\boldsymbol{z}_k, \boldsymbol{\theta}_k)$
  $\boldsymbol{\theta}_{k+1} \leftarrow \max\Big(\boldsymbol{\theta}_k + \delta_k \, \nabla_{\boldsymbol{\theta}} \log p(\widetilde{\mathcal{S}}_k|\boldsymbol{z}_k, \boldsymbol{\theta}_k), 0\Big)$
  $k \leftarrow k + 1$
until convergence

Output: $\boldsymbol{z}$, $\boldsymbol{\theta}$

3.5 Experimental Results

We perform two sets of experiments. First, we use synthetic data to show that, despite the non-smoothness and non-convexity of (3.4), our approach can accurately recover the excitation matrix of the Hawkes process and significantly outperform the classic ML estimator. We further investigated the effects of dimensionality d and the scale of the noise on the performance of our estimator. Second, we validate our approach using a dataset of neuronal spike trains obtained from measurements of the motor cortex of a monkey.

3.5.1 Experiments on Synthetic Data

We set the exponential decay to $\beta = 1$. For smoothing, we use $\beta' = 50$ and $\gamma = 500$, which were found to work well in practice. For each experiment, we choose small positive

[4] Source code of the algorithm is available publicly.

[5] The derivation of the gradient with respect to the noise parameters and the parameters of the Hawkes process is provided in the Appendix.

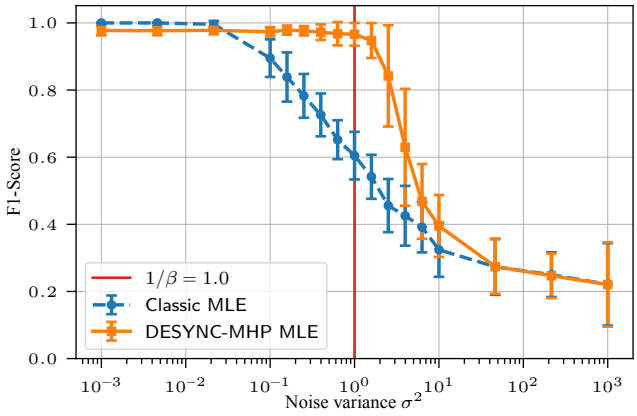


Figure 3.5 – Analysis of the sensitivity to the noise scale with 4 different noise regimes. ($d = 10$ is fixed.)

background intensities $\{\mu_i\}$ and generated a random[6] excitation matrices with entries $\{w_{i,j}\} \in \{0, 1\}$ by sampling edges randomly with probability $2/d$. The average in-degree and out-degree of each node is hence close to 2. We then rescale the entries to obtain a spectral radius of 0.95 to ensure that the simulated processes are stable. We generate $C = 5$ realizations of $50,000$ samples from the Hawkes process using Ogata's thinning algorithm[7] [94]. We repeat each experiment 10 times over 10 different matrices for each set of parameters. We solve the optimization problem (3.4) using stochastic gradient descent with Lasso regularization on the parameters $\{w_{i,j}\}$. We compare our approach, denoted by DESYNC-MHP MLE, against the state-of-the-art maximum likelihood estimation method ADM4 from Zhou et al. [141], denoted by Classic MLE, which solves the classic maximum likelihood estimation problem with the same regularization.

Similar to Chapter 2, we zero-out small weights using a small threshold[8] $\eta = 0.04$ and we report the F1-score of correctly identified edges, *i.e.*, the non-zero the kernels in the support of the excitation matrix.

Sensitivity to the noise level σ^2. We first study the sensitivity of our approach to the level of noise and compared it to the classic ML estimator. Figure 3.5 shows the mean F1-score (and standard deviation) for difference noise variance σ^2. We observe four different noise regimes:

1. In the low-noise regime, virtually no event is swapped, meaning that the cause (parent) events always occur before their effect. Both the classic ML estimator and our approach therefore recover the causal structure accurately.

[6] Experiments were performed on other random graph models with qualitatively similar results.
[7] We used the Python library *tick* to generate synthetic samples of the processes [12].
[8] We provide similar plots for varying thresholds η in Appendix B.2.

2. When the noise level is increased to $\sigma^2 = 1/\beta = 1$ (indicated by the red vertical line in Figure 3.5), our approach still recovers the true causal structure with a F1-score close to 1, contrary to the classic ML estimator whose F1-score drops to 0.6.

3. In the third regime, for noise levels between $\sigma^2 = 1/\beta$ up to one order of magnitude larger than $1/\beta$, our approach gets trapped in local optima more frequently, and hence its performance decreases. Yet, it still clearly outperforms the classic ML estimation.

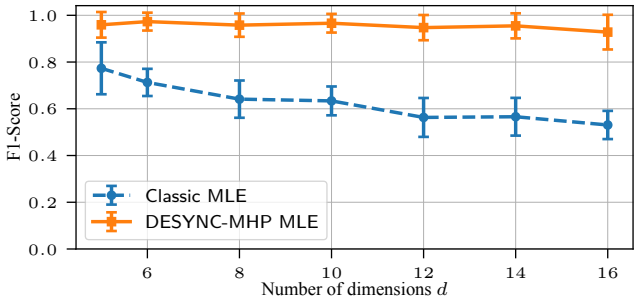
4. In the high-noise regime, the signal from the Hawkes process gets completely lost in the noise. The log-likelihood function therefore rapidly decreases around the true noise $\boldsymbol{z}_*$ and becomes more and more flat for all $\boldsymbol{z}$ far from $\boldsymbol{z}_*$. Thus, iterative gradient-based algorithms such as Algorithm 3.1 and the classic ML estimator stay trapped around their initial points $\boldsymbol{z}_0$. Note that our algorithm with fixed $\boldsymbol{z} = \mathbf{0}$ becomes the classic ML algorithm. As the noise variance increases, neither of the two estimators is able to correctly learn the causal structure in the observations, and both algorithms converge toward sparser excitation matrices. More details are given in the Appendix.

Sensitivity to the number of dimensions d. Because $\boldsymbol{z} \in \mathbb{R}^d$ and $\boldsymbol{\theta} \in \mathbb{R}^{d^2+d}$, number of parameters to estimate grows quadratically with the dimensionality of the process. Consequently, the optimization problem becomes harder for larger-sized problems. In Figure 3.6, we also analyze the sensitivity of our approach to the number of dimensions d of the Hawkes process. We see that our approach still outperforms the Classic MLE as we increase the number of dimensions d.

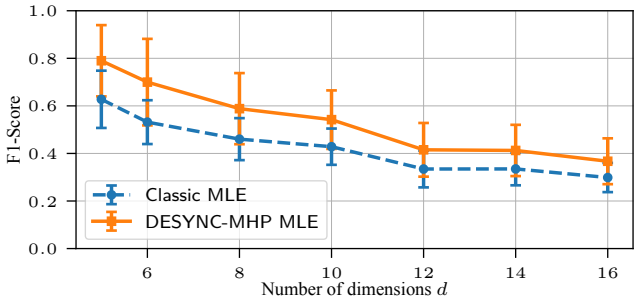
Sensitivity to the number of realizations C. Recall that we used SGD in order to evade local minima in the conditional log-likelihood function. Figure 3.7 shows that with only $C = 3$ independent mini-batches each consisting of $50,000$ samples suffice to obtain a F1-score close 1.

3.5.2 Application to Real Data

In addition to simulations on synthetic data, we also evaluate our approach on an experimental dataset of neuronal spike trains from Wu and Hatsopoulos [126]. The dataset consists in measurements of an electrode array located on the motor cortex of a macaque monkey performing a series of tasks involving a specific arm movement. The local field potentials in the motor cortex were recorded and processed to obtain the neuronal spike train data (discrete event times). More details can be found in [126]. The dataset contains the spike train data from 115 identified neurons for a duration of an hour, quantized at the resolution of 1 millisecond. Since each spike train was recorded



(a) With a noise variance $\sigma^2 = 1$.



(b) With a noise variance $\sigma^2 = 5$.

Figure 3.6 – Analysis of the sensitivity to the number of dimensions for two values of noise variance $\sigma^2 = 1$ and $\sigma^2 = 5$.

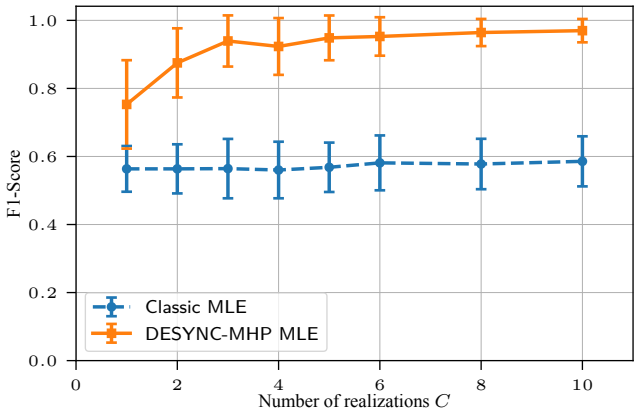


Figure 3.7 – Analysis of the sensitivity to the number of realizations. The dimension $d = 10$, and the noise variance $\sigma^2 = 1$ are fixed.

Table 3.1 – Predictive log-likelihood for the models learned by both approaches. Results are reported averaged over several random initialization points (± standard deviation).

Classic MLE	DESYNC-MHP MLE
$0.4282 \pm 3.5\text{e}{-5}$	$\mathbf{0.4311 \pm 3.0\text{e}{-4}}$

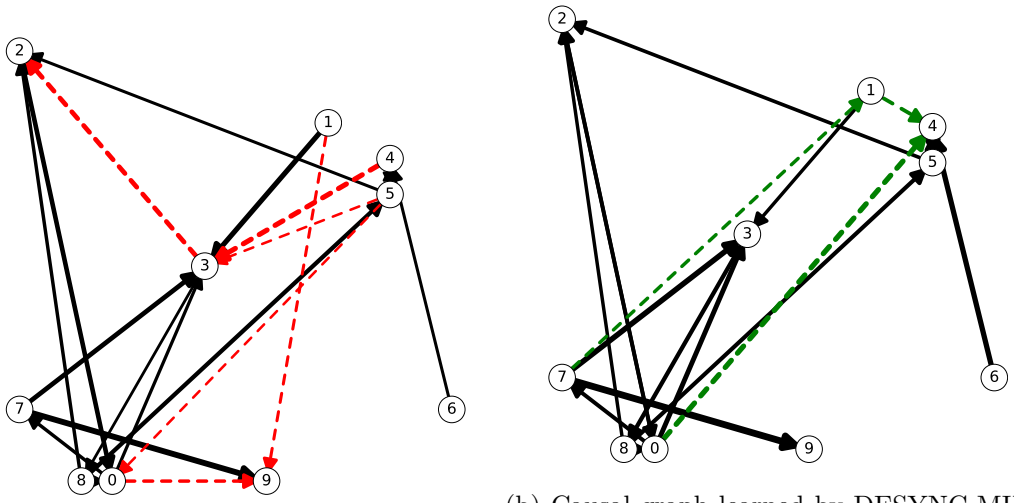
by an independent sensor, some synchronization noise between the dimensions could be expected. For ease of visualization, we keep only a subset of data containing the top $d = 10$ neurons with the highest number of spikes, leading to a total of 354 285 spikes. We use the first 70% of the dataset for training and keep the last 30% for testing. We set the hyperparameters (β, β', γ) to $(0.0047, 0.16, 1.6)$ using grid-search.

We compare the predictive log-likelihood on the test set for the models learned by the baseline classic ML estimator and the DESYNC-MHP ML estimator in Table 3.1. Since problem (3.4) is non-convex, we start the optimization from several starting points and we report both the average and standard deviation of both estimators.

We see that the DESYNC-MHP ML estimate consistently improves the predictive log-likelihood over the classic ML estimate. Our algorithm identifies a small synchronization noise with an average value of 12.5ms, which is less than the average inter-event time of 88.9ms. The Granger causality graphs learned by the two methods is shown in Figure 3.8. The two graphs agree on 91% of the edges. In a previous analysis of causality of the dataset, Quinn et al. [99] identified a dominant direction of influence on both graphs from the lower left to the upper right corner of the array, which might correspond to the direction of propagating local field potential waves discussed in Wu and Hatsopoulos [126]. The Granger causality graphs in Figure 3.8 are consistent with these findings. A dominant direction is indeed noticeable on both graphs and is particularly striking on the graph learned by DESYNC-MHP MLE in Figure 3.8b.

To evaluate the robustness of our approach to larger synchronization noise, we added additional shifts the arrivals in different dimensions randomly with various noise variances σ^2 and computed the predictive log-likelihood both for our algorithm and for the classic ML estimator. The results are reported in Figure 3.9. We identify different noise regimes. For low noise, with a variance smaller than $\sigma^2 = 10$ms, DESYNC-MHP MLE consistently leads to more likely estimate than the classic MLE. This is consistent with the log-likelihood values computed in Table 3.1. For higher noise variance, the likelihood of both approaches decreases, but the DESYNC-MHP ML estimate always outperforms the classic one. It is interesting to note that, on this dataset, the shift in noise regime occurs before $1/\beta$. This might come from the noise initially present in the data.

Although our approach shows better results compared to the classic MLE, the gains are not as large as in the case of the synthetic experiments. Since our approach is not limited to the exponential kernel, results could certainly be improved by using a more flexible



(a) Causal graph learned by the classic MLE. MLE.

(b) Causal graph learned by DESYNC-MHP

Figure 3.8 – Granger causality graphs of the neuronal spike train dataset. Each node indicates a different neuron. The relative position of the nodes corresponds to the relative position of the electrode on the array. The differences between the two graphs is highlighted with dashed edges. Edges appearing only in the classic ML estimate are highlighted in red in Figure 3.8a, and edges appearing only in the DESYNC-MHP ML estimate are highlighted in green in Figure 3.8b. The labels of the nodes correspond to the ordering of the neurons sorted by number of observed events.

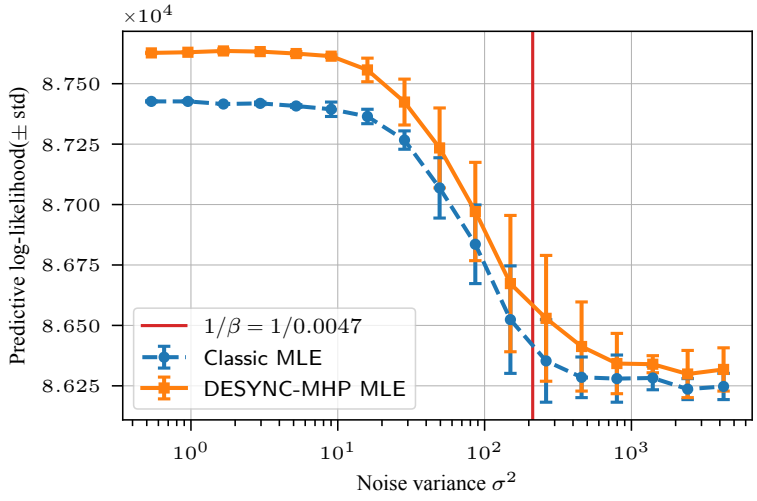


Figure 3.9 – Analysis of the sensitivity to the noise scale on the neuronal spike train dataset.

form of excitation function. For instance, using non-parametric learning approaches for Hawkes processes inspired by Zhou et al. [142], Yang et al. [133] might better fit the true excitation dynamics of the neurons.

3.6 Summary

We addressed the problem of learning the causal structure of multivariate Hawkes processes under synchronization noise, which can arise both for technical reasons or as a feature of the observation process. We showed that the classic maximum likelihood (ML) estimator fails when observations are noisy, because delays perturb the order of events across dimensions. In particular, we showed that, even with small noise with variance $\sigma^2 \approx 1/\beta$, the classic ML estimator is unreliable and only achieves an F1-score of approximately 0.6. To tackle these challenges, we introduced a novel multivariate point process, called DESYNC-MHP, which is a Hawkes process with synchronization noise. In particular, a DESYNC-MHP with parameters $(\boldsymbol{z}, \boldsymbol{\theta})$ is a Hawkes process with parameters $\boldsymbol{\theta}$, where each dimension i is affected by the synchronization noise offset z_i. The log-likelihood function of DESYNC-MHP is non-smooth and non-continuous with respect to the noise, making off-the-shelf gradient-based approaches infeasible. We introduced a novel smoothing approach based on a smooth approximation of the excitation kernels, in conjunction with SGD, to solve the problem. Our experimental results show that, despite the non-convexity of the objective, our approach significantly outperforms the classic ML estimator and accurately recovers the causal structure of Hawkes processes for a wide range of noise.

4 Learning Hawkes Processes under Random Translations

In this chapter[1], we introduce a general class of noise for temporal event data. In this framework, called *random translations*, the observed events in a dimension are subject to random and unknown time shifts that are drawn from some unknown probability distribution. The synchronized noise discussed in Chapter 3 can be seen as a particular type of random translation. In this work, we prove that the cumulants of Hawkes processes are invariant to random translations and hence can be used to learn their underlying causal structure. Furthermore, we empirically characterize the effect of random translations on state-of-the-art learning methods. We show that maximum likelihood-based estimators are brittle, whereas cumulant-based estimators remain stable even in the presence of significant time shifts.

4.1 Introduction

The process through which sequences of events are collected often introduces noise in the observed timestamps. This is particularly relevant for applications relying on data collected from sensors. For instance, in neuroscience, the activity of neurons is typically collected by measuring a continuous signal coming from the action potential of neurons using electrode micro-arrays. The signal is then converted into a discrete sequence of events of firing neurons, called spike trains, which are the times when the action potential exceeds a threshold. This procedure is inherently noisy and prone to introduce inaccuracies in the measured timestamps. Another example is in epidemiology, where the reported times of infection have an approximate granularity and do not account for the latent incubation period. This could lead to inaccuracies in the measured timestamps. As a result, a secondary case might be reported before the primary case, which could interfere with learning the true causation structure.

[1] This chapter is based on Trouleau et al. [117].

Most of the literature on learning temporal point processes assumes perfect information regarding the observation. In this work, we consider inferring the causal network of Hawkes processes when the observations are subject to a particular form of noise, called *random translation*. In a randomly translated point process, every event within a dimension is shifted randomly and independently in time, according to a fixed but unknown distribution. We show that the cumulants of a Hawkes process are invariant with respect to random translations. Therefore, any inference method that can obtain the causal network of a Hawkes process from its cumulants can also be used to learn its causal network under random translation noise.

Outline of the Chapter. We begin by discussing the related works in Section 4.2. In Section 4.3 we define some notations specific to this chapter. In Section 4.4, we introduce the random-translations noise framework. We then characterize the cumulants of a randomly-translated Hawkes processes in Section 4.5, and we discuss the robustness of cumulant-based estimators for learning their excitation matrix in Section 4.6. Finally, we validate our findings with experiments on both synthetic and real data in Section 4.7.

4.2 Related Works

Thanks to its ability to capture the Granger causality between several types of events, the excitation matrix of a Hawkes process has been the target of a number of recent learning algorithms [2, 107, 130, 133]. The main approaches for this problem are of two flavors: maximum likelihood-based approaches [97, 107, 116, 130, 133, 142]; or moment-based approaches that learn the parameters of interest by solving a set of equations obtained from first, second, or third-order moments of the process [2, 8, 7, 44, 58]. For a detailed discussion of these approaches, we refer the reader to Section 1.4. All the aforementioned approaches assume that the observations are noiseless, that is to say, the arrival times of the events are accurately recorded without any delay.

In the previous chapter, we addressed the case where events are synchronized. This is a special case of the random-translation framework that we study in this work. More precisely, in our general random-translation noise model, the events of a dimension are independently shifted according to some unknown distribution. In the synchronized noise model, all events within a dimension have the exact same delay.

The inference of temporal point processes in the presence of noisy observations has been studied for other types of point processes, such as spatial Poisson processes [15, 32]. However, these studies focus mostly on the special case of independent and known noise. Another line of research tackles the inference problem in Hawkes processes with missing data, such as the studies by Shelton et al. [111] and Xu et al. [131]. In our setting, data are not missing, but timestamps are inaccurately measured. In this setting, Hoffmann

and Caramanis [59] consider a similar type of temporal noise in the context of disease modeling. In particular, they study the inference of epidemic pathways for a discrete-time epidemic model spreading over a network of individuals, when the infection times are not known exactly. However, the approaches developed in this work are designed for a discrete-time model where each dimension can have at most only one event, *i.e.*, the infection time of an individual. Hence, these methods are not applicable to our setting. In the context of univariate Hawkes processes, Deutsch and Ross [37] have studied a similar type of noise, referred to as "*data distortion*". They propose an approach to estimate the parameters of the process based on Approximate Bayesian Computation (ABC) and Markov Chain Monte Carlo. However, the method is limited to the univariate setting.

4.3 Preliminaries

We begin by introducing specific notation used throughout the chapter. We denote the Dirac function by $\delta(t)$. For a given function $f(t)$, we denote its time reversed version

$$\underline{f}(t) \coloneqq f(-t),$$

and we define its convolution with a function $g(t)$ by

$$f * g(t) \triangleq \int_{\mathbb{R}} f(t-x)g(x)dx.$$

We use $f^{*n}(t)$ to denote the convolution of $f(t)$ with itself n times. The n-dimensional Laplace transform of a function $f(\boldsymbol{x})$ is given by

$$\mathcal{L}[f](\boldsymbol{s}) \triangleq \int_{\mathbb{R}^n} f(\boldsymbol{x}) \exp(-\boldsymbol{s}^T\boldsymbol{x})d\boldsymbol{x}.$$

Finally, the Laplace transform of a matrix function $\boldsymbol{\Phi}(t) = [\phi_{i,j}(t)]$, denoted by $\mathcal{L}[\boldsymbol{\Phi}](s) \triangleq [\mathcal{L}[\phi_{i,j}](s)]$, is done element-wise.

Recall that, as defined in Section 1.3, a d-dimensional Hawkes process is a collection of d univariate temporal point processes $N_i(t)$, $i = 1, \ldots, d$, also called dimensions, with a conditional intensity function given in (1.18). In this chapter, the *integrated* excitation matrix

$$\boldsymbol{\Phi} \triangleq \mathcal{L}[\boldsymbol{\Phi}](0)$$

is the quantity we are interested in estimating.

4.4 Random Translation Noise Framework

In a randomly translated point process, all the events are shifted randomly in time, according to an unknown distribution [34]. More precisely, consider a sequence of events

$$\mathcal{S} = \{(i_n, t_n)\}_{n\geq 1}, \tag{4.1}$$

where t_n denotes the time of the n-th event and i_n is its dimension. A random translation of $\mathcal{S}$ is denoted by $\widetilde{\mathcal{S}}$ and is defined by

$$\widetilde{\mathcal{S}} = \{(i_n, \tilde{t}_n)\}_{n\geq 1} \coloneqq \{(i_n, t_n + x_n)\}_{n\geq 1}, \tag{4.2}$$

where $\{x_n\}_{n\geq 1}$ are independent random variables such as $x_n \sim F_{i_n}(\cdot)$. Namely, for each type $i \in [d]$, the timestamps of type i events are shifted independently with distribution $F_i(\cdot)$. Figure 4.1 demonstrates a simple Hawkes process in three dimensions, in which events are translated according to distribution functions $\{F_A, F_B, F_C\}$. Note that the synchronization noise model proposed in Chapter 3 is a special case of the random translation, when all the distributions are Dirac delta functions, *i.e.*, for every i, $dF_i(x) = \delta(x - z_i)dx$, where $z_i \in \mathbb{R}_+$.

Among the potential approaches to learning randomly translated Hawkes processes, a first candidate is a maximum-likelihood based estimation, such as expectation maximization. However, as discussed in Chapter 3, such a method results in a non-convex objective function, has a high computational complexity, and fails –even for the synchronized translations– as the noise power increases. For the sake of completeness, we will demonstrate the similar shortcomings of the maximum-likelihood estimator for the random-translation setting through empirical experiments.

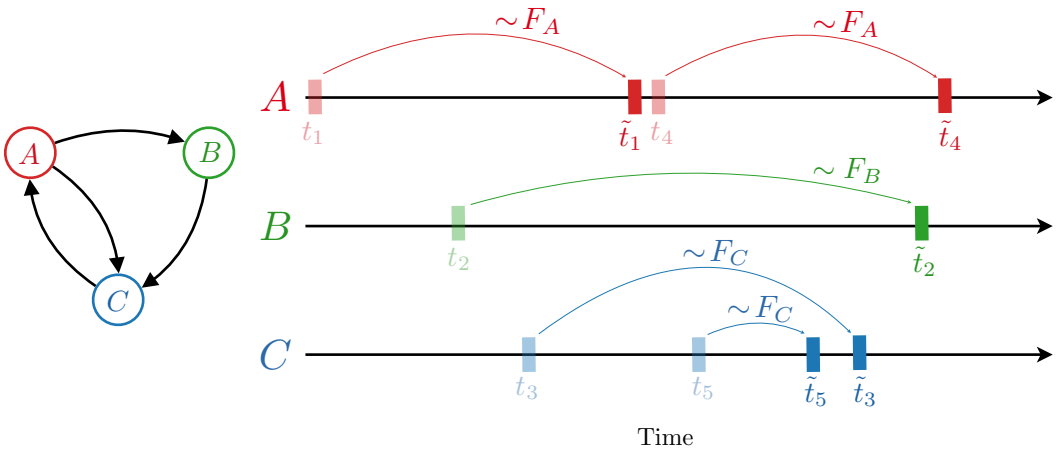


Figure 4.1 – An example of events in a three-dimensional Hawkes process and their translations. Events in dimension A, (resp., B and C) are translated randomly by F_A (resp., F_B and F_C). The Granger causality graph of the process is shown on left.

4.5 Cumulants of Randomly Translated Hawkes Process

As discussed in Section 1.3.3, Jovanović et al. [62] showed that the cumulant densities of the Hawkes process can be calculated analytically through their cluster representation. This result establishes the relationships between the integrated cumulants of a Hawkes process and its excitation matrix. Achab et al. [2] used this relationship to develop an algorithm called NPHC to learn the causal network of a Hawkes process given its integrated cumulants. They also provided an estimator for the first, second, and third-order integrated cumulants given a set of observations.

In this section, we will compute the cumulant densities of a randomly translated Hawkes process by using its cluster representation and show how they relate to the causal structure of the underlying process. To do so, we have to study the effect of random translations on the clusters of a Hawkes process. We observe two key properties that we discuss in the context of a simple example illustrated in Figure 4.2.

As shown in the figure, although the events within this cluster are randomly displaced, the tree structure –*i.e.*, the parent-children relationships– of the cluster is unaffected. Moreover, the clusters do not mix, *i.e.*, two separate clusters remain separated after translation. The next theorem follows from these properties and expresses the cumulant densities of a randomly translated Hawkes process as functions of the translation distributions and the parameters of the process.

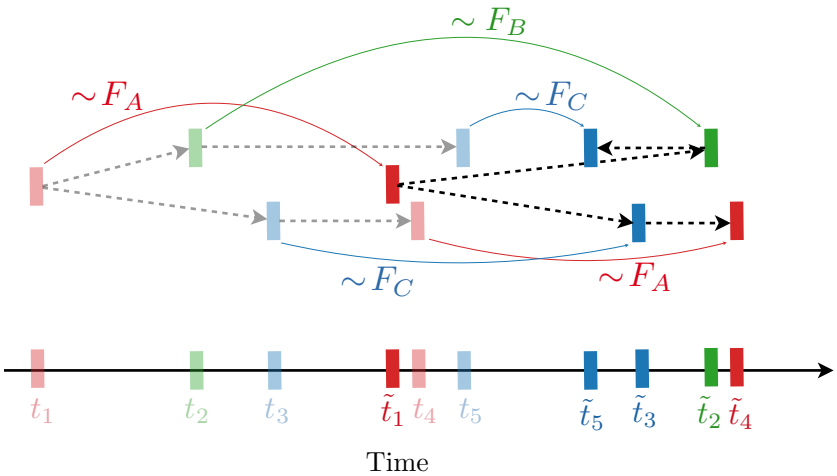


Figure 4.2 – The cluster of Figure 4.1, with the immigrant of type A and its four descendants translated according to distributions $\{F_A, F_B, F_C\}$.

Theorem 4.1. *Consider a Hawkes process with excitation matrix function $\boldsymbol{\Phi}(t)$ and exogenous intensity vector $\boldsymbol{\mu} \in \mathbb{R}_+^d$. After a random translation of the event set $\mathcal{S}$ with distributions $\{F_1(\cdot), \dots, F_d(\cdot)\}$, the resulting event set $\widetilde{\mathcal{S}}$ has the following cumulants.*

$$K_i = \sum_{m=1}^{d} \mu_m \int_{\mathbb{R}} \widetilde{R}_{i,m}(x)dx, \tag{4.3}$$

$$K_{i,j}(\tilde{t}_1, \tilde{t}_2) = \sum_{m=1}^{d} K_m \int_{\mathbb{R}} \widetilde{R}_{i,m}(\tilde{t}_1 - x)\widetilde{R}_{j,m}(\tilde{t}_2 - x)dx, \tag{4.4}$$

$$\begin{aligned}
&K_{i,j,k}(\tilde{t}_1, \tilde{t}_2, \tilde{t}_3) = \\
&\quad \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} \Big(\widetilde{R}_{i,n}(\tilde{t}_1 - x)\widetilde{R}_{j,m}(\tilde{t}_2 - y)\widetilde{R}_{k,m}(\tilde{t}_3 - y)\widetilde{\Psi}_{m,n}(y - x)\Big)dydx \\
&+ \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} \Big(\widetilde{R}_{j,n}(\tilde{t}_2 - x)\widetilde{R}_{i,m}(\tilde{t}_1 - y)\widetilde{R}_{k,m}(\tilde{t}_3 - y)\widetilde{\Psi}_{m,n}(y - x)\Big)dydx \\
&+ \sum_{m,n=1}^{d} K_n \iint_{\mathbb{R}} \Big(\widetilde{R}_{k,n}(\tilde{t}_3 - x)\widetilde{R}_{i,m}(\tilde{t}_1 - y)\widetilde{R}_{j,m}(\tilde{t}_2 - y)\widetilde{\Psi}_{m,n}(y - x)\Big)dydx \\
&+ \sum_{m=1}^{d} K_m \int_{\mathbb{R}} \Big(\widetilde{R}_{i,m}(\tilde{t}_1 - x)\widetilde{R}_{j,m}(\tilde{t}_2 - x)\widetilde{R}_{k,m}(\tilde{t}_3 - x)\Big)dx,
\end{aligned} \tag{4.5}$$

where $\widetilde{\boldsymbol{R}}(t) \coloneqq \sum_{n\geq 0} \widetilde{\boldsymbol{\Phi}}^{*n}(t)$, $\widetilde{\boldsymbol{\Psi}}(t) \coloneqq \widetilde{\boldsymbol{R}}(t) - \boldsymbol{I}\delta(t)$, *and*

$$\begin{aligned}
\widetilde{\phi}_{i,j}(t) &= f_i * \phi_{i,j} * \underline{f}_j(t) \\
&= \iint_{\mathbb{R}} f_i(t + x - s)\phi_{i,j}(s)f_j(x)dsdx.
\end{aligned} \tag{4.6}$$

with $f_i(x)dx = dF_i(x)$.

We prove the theorem in Appendix A.3.1. The steps of the proof can be summarized as follows. First, we show that an inhomogeneous Poisson process is another inhomogeneous Poisson process after random translation. Then, using this fact, along with the Poisson cluster representation of the Hawkes process discussed in Section 1.3.1, we determine the intensity function of each cluster in order to derive the equations of the cumulants.

This result shows the relationships between the first, second, and third-order cumulant densities of a randomly translated Hawkes process, the noise distributions, and the parameters of the underlying process. Note that Equation (4.6) implies that the matrices $\boldsymbol{\Phi}(t)$ and $\widetilde{\boldsymbol{\Phi}}(t)$ have the same support. In the next corollary, we further show that their integrated versions, namely $\boldsymbol{\Phi} \coloneqq \mathcal{L}[\boldsymbol{\Phi}](0)$ and $\overline{\boldsymbol{\Phi}} \coloneqq \mathcal{L}[\widetilde{\boldsymbol{\Phi}}](0)$, are equal.

Corollary 4.2. *Consider a Hawkes process with stationary increments. After a random translation, its corresponding matrix function $\widetilde{\boldsymbol{R}}(t)$ given in Theorem 4.1 is bounded, and*

$$\overline{\boldsymbol{R}} = \left(\boldsymbol{I} - \overline{\boldsymbol{\Phi}}\right)^{-1}, \tag{4.7}$$

$$\overline{\boldsymbol{\Phi}} = \boldsymbol{\Phi}, \tag{4.8}$$

where $\overline{\boldsymbol{R}} \coloneqq \mathcal{L}[\widetilde{\boldsymbol{R}}](0)$, and $\overline{\boldsymbol{\Phi}} \coloneqq \mathcal{L}[\widetilde{\boldsymbol{\Phi}}](0)$.

We use this equivalence to learn the support of $\boldsymbol{\Phi}$. Note that, given a realization $\widetilde{\mathcal{S}}$ of a randomly translated Hawkes process, we can empirically estimate the integrated cumulants. In the remainder of this section, we transform the equations (4.3)-(4.5) into their integrated forms by evaluating their Laplace transform at $s = 0$ and solve for $\overline{\boldsymbol{R}}$. Corollary 4.2 can then be applied to obtain $\boldsymbol{\Phi}$. More precisely, let

$$\begin{aligned}
\overline{\Psi}_{i,j} &= \mathcal{L}[\widetilde{\Psi}_{i,j}](0), \\
\overline{K}_{i,j} &= \mathcal{L}[K_{i,j}](0), \\
\overline{K}_{i,j,k} &= \mathcal{L}[K_{i,j,k}](0).
\end{aligned}$$

Then, the integrated cumulants of a randomly translated Hawkes process can be computed from (4.3)-(4.5) as follows.

$$K_i = \sum_{m=1}^{d} \mu_m \overline{R}_{i,m}, \tag{4.9}$$

$$\overline{K}_{i,j} = \sum_{m=1}^{d} K_m \overline{R}_{i,m} \overline{R}_{j,m}, \tag{4.10}$$

$$\begin{aligned}
\overline{K}_{i,j,k} = &\sum_{m,n=1}^{d} K_n \overline{R}_{i,n} \overline{R}_{j,m} \overline{R}_{k,m} \overline{\Psi}_{m,n} \\
&+ \sum_{m,n=1}^{d} K_n \overline{R}_{j,n} \overline{R}_{i,m} \overline{R}_{k,m} \overline{\Psi}_{m,n} \\
&+ \sum_{m,n=1}^{d} K_n \overline{R}_{k,n} \overline{R}_{i,m} \overline{R}_{j,m} \overline{\Psi}_{m,n} \\
&+ \sum_{m=1}^{d} K_m \overline{R}_{i,m} \overline{R}_{j,m} \overline{R}_{k,m},
\end{aligned} \tag{4.11}$$

where $\overline{\boldsymbol{\Psi}} = \overline{\boldsymbol{R}} - \boldsymbol{I}$.

We emphasize that the above equations are analogous to those of a Hawkes process without random translations given in Section 1.3.3. Together with the fact that $\overline{\boldsymbol{\Phi}} = \boldsymbol{\Phi}$, this implies that the integrated cumulants are invariant with respect to random translations, a key result that will enable to estimate them consistently.

In Equations (4.9)-(4.11), the first-order cumulants $\{K_i\}$ and the integrated cumulants $\{\{\overline{K}_{i,j}\}, \{\overline{K}_{i,j,k}\}\}$ can be empirically estimated from the data. These estimates are then used to solve for $\overline{\boldsymbol{R}} = [\overline{R}_{i,j}]$, which yields the underlying causal structure, *i.e.*, the support of $\boldsymbol{\Phi}$, of the randomly translated Hawkes process, via Corollary 4.2. In the next section, we review two approaches for learning Hawkes processes based on their cumulants and show how exactly they can be adopted to infer the underlying causal structures of randomly translated Hawkes processes.

4.6 Cumulant-Based Estimation Methods

4.6.1 The NPHC Algorithm

Achab et al. [2] proposed the NPHC algorithm, a non-parametric approach inspired by the generalized method of moments. First, note that (4.10) and (4.11) provide $(d^2 + d)/2$ and $(d^3 + 3d^2 + 2d)/6$ independent equations, respectively. The number of unknowns, $\{\{\mu_i\}_{i=1}^d, \overline{\boldsymbol{R}}\}$, is only $d + d^2$. Achab et al. [2] then select a subset of size d^2 equations out of the group of equations in (4.11), namely, $\overline{K}_{i,i,j}$ for $1 \leq i, j \leq d$, and use $d^2 + (d^2 + d)/2$ equations to obtain the unknowns. The NPHC algorithm works in two steps.

(1) First, the integrated cumulants are estimated from the data. Let

$$\widehat{\overline{\boldsymbol{C}}} \coloneqq \left[\widehat{\overline{K}}_{i,j}\right] \text{ and } \widehat{\overline{\boldsymbol{S}}} \coloneqq \left[\widehat{\overline{K}}_{i,i,j}\right]$$

denote the estimators of the integrated covariance matrix and skewness matrix, respectively. Details of these estimators are provided in Appendix A.3.4.

(2) Then, the NPHC estimator for $\overline{\boldsymbol{R}}$ is defined as the solution of a polynomial optimization problem

$$\widehat{\overline{\boldsymbol{R}}} \in \arg\min_{\overline{\boldsymbol{R}}} \ (1-\alpha)\|\boldsymbol{S}(\overline{\boldsymbol{R}}) - \widehat{\overline{\boldsymbol{S}}}\|_2^2 + \alpha\|\boldsymbol{C}(\overline{\boldsymbol{R}}) - \widehat{\overline{\boldsymbol{C}}}\|_2^2.$$

The weight $\alpha = \|\widehat{\overline{\boldsymbol{S}}}\|_2^2 \Big/ \left(\|\widehat{\overline{\boldsymbol{S}}}\|_2^2 + \|\widehat{\overline{\boldsymbol{C}}}\|_2^2\right)$ balances between the two terms matching the integrated covariance matrix $\boldsymbol{C}(\overline{\boldsymbol{R}}) = \left[\overline{K}_{i,j}\right]$ and the integrated skewness matrix $\boldsymbol{S}(\overline{\boldsymbol{R}}) = \left[\overline{K}_{i,i,j}\right]$.

The authors prove that the NPHC estimator is consistent[2]. Corollary 4.2 then demonstrates that the NPHC estimator is also consistent for randomly translated Hawkes processes. Therefore, applying the NPHC algorithm to a randomly translated sequence of events will recover the matrix $\overline{\boldsymbol{R}}$ and, consequently, the integrated excitation matrix $\boldsymbol{\Phi}$.

[2]For more comprehensive details on the algorithm and its relation to the generalized method of moments, we refer the reader to [2].

4.6.2 The Wiener-Hopf Formulation

Another cumulant-based approach for learning Hawkes processes is based on the second-order statistics [7]. More precisely, we define the covariance density matrix of a Hawkes process, $\mathbf{\Sigma}(t_1, t_2) = [\Sigma_{i,j}(t_1, t_2)]$ as

$$\Sigma_{i,j}(t_1, t_2) \coloneqq K_{i,j}(t_1, t_2) - \frac{\mathbb{E}[dN_i(t_1)]}{dt_1}\epsilon_{i,j}\delta(t_1 - t_2),$$

where $\epsilon_{i,j}$ is the Kronecker symbol, which is always 0 except when $i = j$, in which case it is 1. Hawkes [58] proved that $\mathbf{\Sigma}(t) \coloneqq \mathbf{\Sigma}(t, 0)$ is directly related to the excitation matrix $\mathbf{\Phi}(t)$ through the equation

$$\mathbf{\Sigma}(t) = \left(\boldsymbol{I}\delta + \underline{\mathbf{\Psi}}\right) * \mathbf{\Lambda}\left(\boldsymbol{I}\delta + \mathbf{\Psi}\right)^T(t) - \mathbf{\Lambda}\delta(t), \ \forall t \in \mathbb{R}, \tag{4.12}$$

where $\mathbf{\Lambda} \coloneqq \mathrm{diag}([K_1, \dots, K_d])$ is the mean intensity of the stationary process, $\mathbf{\Psi}(t) \coloneqq \sum_{n\geq 1} \mathbf{\Phi}^{*n}(t)$ and $\underline{\mathbf{\Psi}}(t) = \mathbf{\Psi}(-t)$. Note that this equation does not admit a unique solution with respect to $\mathbf{\Phi}(t)$.

Bacry and Muzy [7] derived the following d^2-dimensional Wiener-Hopf system of equations from (4.12):

$$\boldsymbol{X}(t) = \mathbf{\Phi}(t) + \mathbf{\Phi} * \boldsymbol{X}(t), \ \forall t > 0, \tag{4.13}$$

where $\boldsymbol{X}(t) = \mathbf{\Sigma}^T(t)\mathbf{\Lambda}^{-1}$ can be estimated from data. The interesting aspect of this equation is that, since it only consider positive times, using the fact that $\mathbf{\Phi}(t)$ is causal results in a unique solution with respect to $\mathbf{\Phi}(t)$. It can therefore be used to infer the excitation matrix $\mathbf{\Phi}(t)$ of a Hawkes process from data.

Similarly to the aforementioned approach, we can use Theorem 4.1 to define the covariance density matrix of a randomly translated Hawkes process and to explicit its relation to $\widetilde{\mathbf{\Phi}}(t)$ which was defined in (4.6).

Corollary 4.3. *Let $\widetilde{\mathbf{\Sigma}}(t)$ denotes the covariance density matrix of a randomly translated Hawkes process, defined as*

$$\widetilde{\Sigma}_{i,j}(\tilde{t}_1, \tilde{t}_2) \coloneqq K_{i,j}(\tilde{t}_1, \tilde{t}_2) - \frac{\mathbb{E}[dN_i(\tilde{t}_1)]}{d\tilde{t}_1}\epsilon_{i,j}\delta(\tilde{t}_1 - \tilde{t}_2).$$

Then, for all $t \in \mathbb{R}$,

$$\widetilde{\mathbf{\Sigma}}(t) = \left(\boldsymbol{I}\delta + \underline{\widetilde{\mathbf{\Psi}}}\right) * \mathbf{\Lambda}\left(\boldsymbol{I}\delta + \widetilde{\mathbf{\Psi}}\right)^T(t) - \mathbf{\Lambda}\delta(t), \tag{4.14}$$

where $\mathbf{\Lambda} = \mathrm{diag}([K_1, \dots, K_d])$ and $\widetilde{\mathbf{\Psi}}(t)$ is defined as in Theorem 4.1.

Similar to (4.12), Equation (4.14) does not necessarily admit a unique solution[3] with respect to $\widetilde{\mathbf{\Phi}}(t)$, but unlike $\mathbf{\Phi}(t)$, $\widetilde{\mathbf{\Phi}}(t)$ is not a causal function. This is evident from (4.6) because $\widetilde{\phi}_{i,j}(t)$ is obtained by convolving the causal function $\phi_{i,j}(t)$ with functions $\{\underline{f}_j(t), f_i(t)\}$ in which at least one is an anti-causal function. This is a major hurdle that was not present in the noiseless case but comes with any non-zero amounts of noise. Indeed, it prevents us from obtaining a Wiener-Hopf system of equations from (4.14) that, like (4.13), admits a unique solution. Nevertheless, for a small amount of noise, experiments show that we can successfully apply the Wiener-Hopf approach from Bacry and Muzy [7] to randomly translated Hawkes processes and learn $\widetilde{\mathbf{\Phi}}(t)$ by solving the system

$$\widetilde{\boldsymbol{X}}(t) = \widetilde{\mathbf{\Phi}}(t) + \widetilde{\mathbf{\Phi}} * \widetilde{\boldsymbol{X}}(t), \ \forall t > 0, \tag{4.15}$$

where $\widetilde{\boldsymbol{X}}(t) = \widetilde{\Sigma}^T(t)\Lambda^{-1}$. However, because $\widetilde{\mathbf{\Phi}}(t)$ increasingly departs from being causal as the noise power increases, this approach fails to accurately learn the underlying causal structure.

4.7 Experimental Results

To illustrate the result of Theorem 4.1 and to characterize the effect of random translations on the estimation of Hawkes processes, we carry out two sets of experiments. First, we simulate a synthetic dataset from a Hawkes process and quantify the ability of two maximum likelihood-based and two cumulant-based approaches for learning the ground-truth excitation matrix, under varying levels of noise power. Second, we evaluate the stability of each approach to random translations on a real dataset pertaining to Bund Future traded at Eurex. The open-source code and datasets used in all experiments are publicly available for reproducibility[4].

We evaluate the effect of random translation on the following four state-of-the-art approaches.

- NPHC. (from Achab et al. [2]) This non-parametric approach is based on matching the empirical integrated cumulants of the events, as discussed in Section 4.6.1.
- WH. (from Bacry and Muzy [7]) This method is a non-parametric approach based on solving a set of Wiener-Hopf equations for learning the excitation functions of the process, as discussed in Section 4.6.2.
- ADM4. (from Zhou et al. [142]) This method is a parametric approach that maximizes the log-likelihood function with a sparse and low-rank regularization. It

[3] We present a proof in Appendix A.3.5.
[4] For more details, see Appendix C.3.

assumes an exponential excitation function of the form $\phi_{i,j}(t) = w_{i,j}\kappa(t)$, where $\kappa(t) = \beta \exp(-\beta t)$. The exponential decay β is a given hyperparameter.

- Desync-MLE. (from Trouleau et al. [116]) This method, discussed in Chapter 3, is the parametric approach that maximizes the log-likelihood function of a Hawkes process under synchronization noise, *i.e.*, a particular type of random translation where the noise is assumed to be distributed as $dF_i(x) = \delta(x - z_i)dx, \ \forall i \in [d]$, such that all events within a dimension are shifted by a constant. This method jointly learns the parameters of the process, as well as the noise value $\{z_i\}$, by using stochastic gradient descent. Similarly to ADM4, this approach assumes exponential excitation functions where the exponential decay β is a given hyperparameter.

4.7.1 Synthetic Data

We first apply the result of Theorem 4.1 to a synthetic 10-dimensional ($d = 10$) Hawkes process. Following the experimental setup of Achab [1], we considered a non-symmetric block-matrix $\mathbf{\Phi}^\star$ depicted in Figure 4.3(a), with exponential excitation functions

$$\phi^\star_{i,j}(t) = w_{i,j}\beta \exp(-\beta t), \forall i, j = 1, \dots, d,$$

with $\beta = 1$, and baseline intensity $\mu_i = 0.01, \forall i = 1, \dots, d$.

We simulated 20 datasets, each comprised of 5 realizations of 10^5 events. We then randomly translated each dataset with distributions $F_i \sim \mathcal{N}(0, \sigma^2), \ 1 \leq i \leq d$, for varying noise powers σ^2, and we estimated the excitation matrix for the aforementioned approaches[5]. All reported values are averaged over the 20 simulated datasets ($\pm$ standard error).

Figure 4.3 depicts the estimated integrated excitation matrices for a fixed noise level $\sigma^2 = 5$ for a qualitative visualization of the results. We observe that, though the cumulant-based NPHC method is able to accurately recover the excitation matrix, the maximum likelihood-based ADM4 suffers from both false positives and misses true positives. The covariance-based WH approach is performing better than ADM4 but tends to suffer from false positives. This is expected from Corollary 4.3, as WH incorrectly assumes that $\widetilde{\mathbf{\Phi}}(t)$ is a causal function.

To verify the findings of Theorem 4.1, we evaluated the sensitivity of the estimators of the integrated cumulants used in NPHC. This pertains to the estimation of the left-hand side of (4.4) and (4.5). In Figure 4.4, we report the squared $L_{2,2}$ distance of the estimated integrated covariance and skewness matrices to their corresponding ground-truth. As expected, the cumulant estimators remain stable over a large range of noise levels.

[5] We also ran experiments with other noise distributions (*i.e.*, exponential and uniform) and observed similar results.

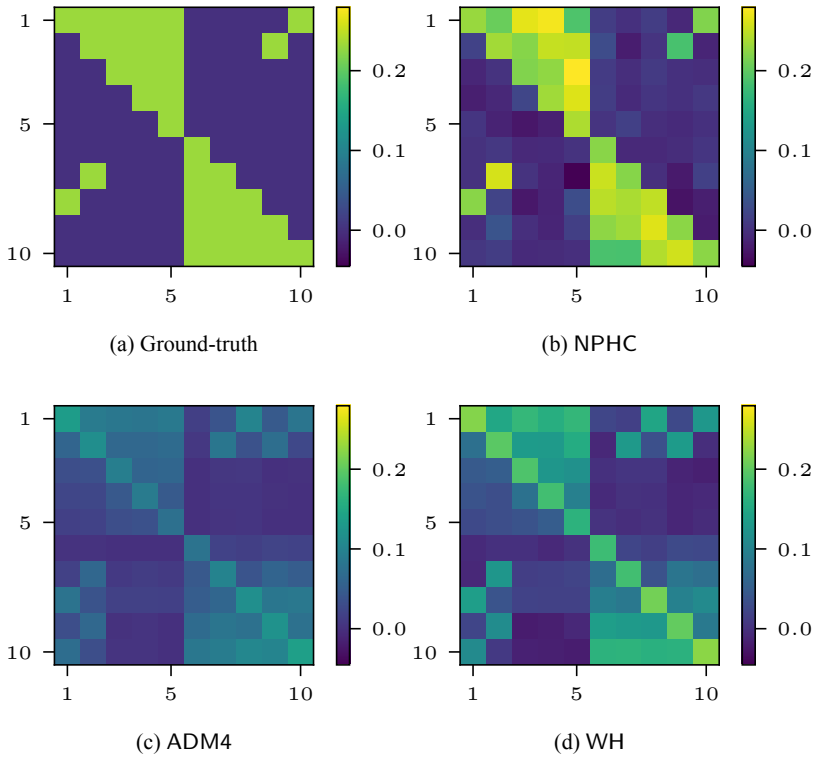


Figure 4.3 – Comparison of estimated integrated excitation matrix $\mathbf{\Phi}$ for several methods under randomly translated observations.

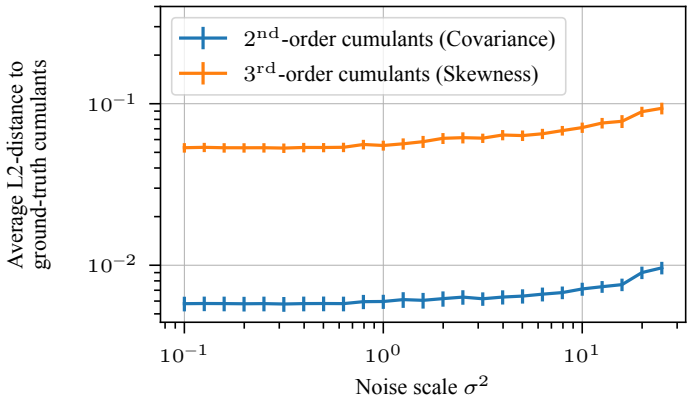


Figure 4.4 – Analysis of sensitivity of integrated cumulant estimation with respect to the scale of the noise.

To further quantitatively evaluate the sensitivity of each approach to increasing noise levels, we also measured their performance against several metrics for a large range of noise variances σ^2. More specifically, we considered the following metrics.

- **Relative error.** To evaluate the distance between the estimated and the ground-truth values, we computed the averaged relative error defined as

$$\begin{cases} |\widehat{\phi}_{i,j} - \phi^\star_{i,j}|/|\phi^\star_{i,j}|, & \text{if } \phi^\star_{i,j} > 0, \\ |\widehat{\phi}_{i,j} - \phi^\star_{i,j}|/\min_{\phi^\star_{m,n}\neq 0} |\phi^\star_{m,n}|, & \text{otherwise.} \end{cases}$$

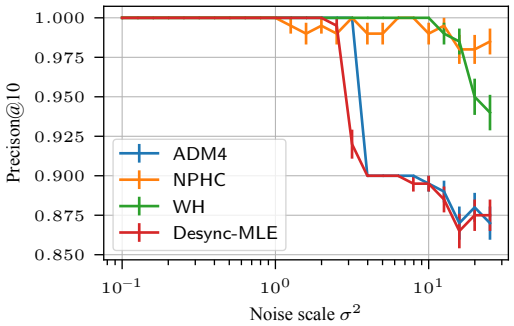
  This metric is more sensitive to errors in small values and therefore penalizes methods with large false positive entries learned in the excitation matrix [49, 142].

- **Precision@k.** To assess the performance of the approaches at recovering the top entries in $\mathbf{\Phi}^\star$, we used precision@k that is defined as the average fraction of correctly identified entries in the top k largest estimated values. We reported this metric for $k = 10$ and $k = 20$ [49, 107].

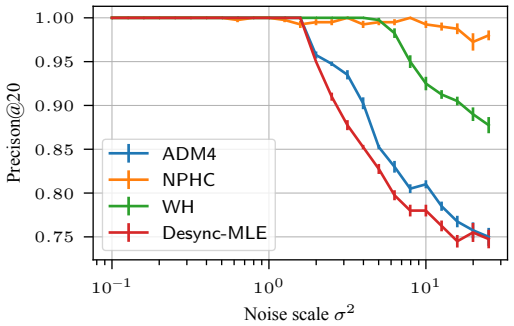
- **PR-AUC.** Considering that there is a Granger-causal link between two dimensions if the learned value $\widehat{\phi}_{i,j} > \eta$, we evaluate the performance of the resulting binary classification problem by using the area under the precision-recall curve over all thresholds $\eta > 0$. Methods that accurately uncover the excitation patterns from the randomly translated data will have a PR-AUC close to 1.

- $L_{2,2}$ **Norm.** We also measured the squared $L_{2,2}$ norm of the estimated excitation matrices, defined as $\|\widehat{\mathbf{\Phi}}\|^2_{2,2} = \sum_{i,j} \widehat{\phi}^2_{i,j}$. Methods that fail to uncover the excitation patterns from the randomly translated data tend to learn an almost-zero matrix with small $L_{2,2}$ norm.

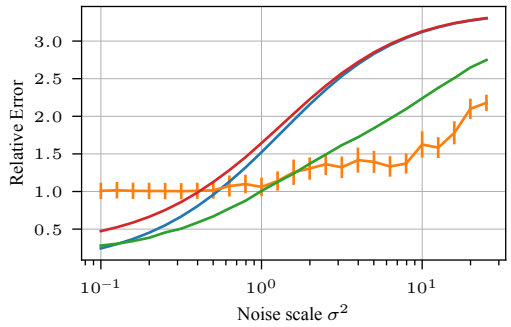
The results are shown in Figure 4.5. As expected from Corollary 4.2, the NHPC estimator provides stable estimates for a large range of noise levels. Whereas Figure 4.5e shows that the norm of the matrices estimated by the other approaches tends to zero with increasing σ^2. This is particularly obvious for ADM4 and Desync-MLE. This result is consistent with the findings of Chapter 3 for the special case of synchronized noise. Consistently with the observation discussed in Section 4.6.2, the WH method performs well only for low noise. This is because, as expected, the non-causal property of $\widetilde{\mathbf{\Phi}}(t)$ in randomly translated Hawkes process violates the assumption of WH and hence introduces a bias in the estimation. In a smaller noise regime, $\widetilde{\mathbf{\Phi}}(t)$ is closer to being causal and, as a result, the WH method learns it better.



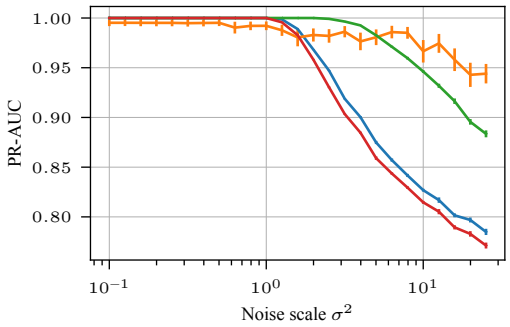
(a) Precision@10



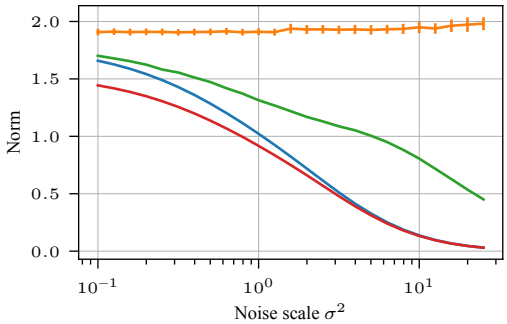
(b) Precision@20



(c) Relative Error



(d) PR-AUC



(e) $L_{2,2}$ Norm

Figure 4.5 – Analysis of the sensitivity of the estimation methods to the noise scale for the synthetic datasets. Reported values are averaged over 20 simulated datasets (± standard error).

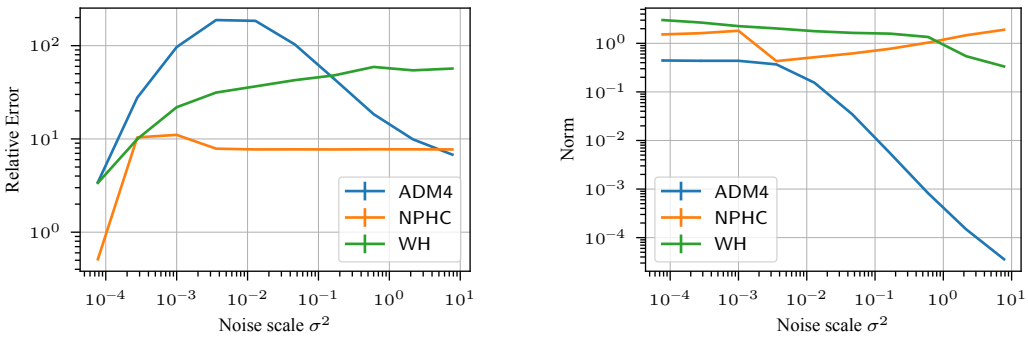
4.7.2 Real Data

We also evaluated the effect of random translations on a publicly available real-world dataset of Bund Futures traded at Eurex over 20 days in April 2014[6]. This dataset contains $d = 4$ dimensions corresponding to the following types of events:

- mid-price movement up,
- mid-price movement down,
- buyer initiated trades that do not move the mid-price,
- seller initiated trades that do not move the mid-price.

As there is no ground-truth available for this dataset, we focus our experiments on evaluating the stability of the estimates when a random translation is added to the observations. More precisely, for a large range of noise levels σ^2, we randomly shifted the observed timestamps with distributions $F_i \sim \mathcal{N}(0, \sigma^2)$, and compared the resulting estimated $\widehat{\mathbf{\Phi}}_\sigma$ to the noise-free estimate $\widehat{\mathbf{\Phi}}_0$ based on the dataset without random translation.

We show the results in Figure 4.6. We observe that they are consistent with the conclusions reached on the synthetic datasets. ADM4 converges to a zero excitation-matrix, as the noise scale increases, whereas the cumulant-based approaches, NPHC and WH, remain stable for a wider range of noise levels.



(a) Relative Error (b) $L_{2,2}$ Norm

Figure 4.6 – Analysis of the sensitivity of the estimation to noise scale for the Bund Futures traded at Eurex. Reported values are averaged over 20 simulated datasets (± standard error).

[6]Dataset available at: `https://github.com/X-DataInitiative/tick-datasets/`

4.8 Summary

In this work, we have studied the inference problem of multivariate Hawkes processes under noisy timestamps. We have introduced a general form of observation noise called random translation and proved that the cumulants of Hawkes processes are invariant to such noise. We derived a set of equations for the first, second, and third-order cumulants of a randomly translated Hawkes process with respect to its underlying parameters, namely, the exogenous intensities and the excitation matrix. Using these findings, we have shown than cumulant-based estimators, such as NPHC, are robust to random translations and can accurately learn the causal structure of randomly translated Hawkes processes. Furthermore, through extensive experiments on both synthetic and real datasets, we validated our results and demonstrated that the state-of-the-art inference methods based on maximum-likelihood fail to capture the structure, when the observations are affected by random translations.

5 Learning Large Networks With Wold Processes

In this chapter[1], we shift our focus from Hawkes processes and consider another class of temporal point process, called *Wold processes*. Although Hawkes processes are certainly the most widely used class of temporal point processes and are responsible for the growing popularity of this type of models, their Bayesian treatment introduces computational challenges that severely limit their scalability. The only approaches proposed to overcome this challenge are based on discrete-time approximations of the model, that incur an information loss. Wold processes address this limitation thanks to the form of their conditional intensity function. In addition to a computational advantage, recent statistical studies showed that Wold processes are well suited to capture the dynamics of real-world communications. In this work, we relax some of the restrictive modeling assumptions made in the state of the art and introduce a Bayesian approach for inferring the parameters of multivariate Wold processes. We develop a computationally efficient variational inference algorithm that allows scaling up the approach to high-dimensional processes and long sequences of observations. Our experimental results on both synthetic and real-world datasets show that our proposed algorithm outperforms existing methods.

5.1 Introduction

Wold processes [34, 125], akin to Hawkes processes, are a type of multivariate point process that are well suited for modeling discrete events. They are defined in terms of a Markovian joint distribution of *inter-event times*. Specifically, the times between consecutive events t_{n-1} and t_n, also called inter-event times $\delta_n \coloneqq t_n - t_{n-1}$, form a Markov chain such that the distribution $p(\delta_n|\delta_{n-1}, \delta_{n-2}, ..., \delta_1) = p(\delta_n|\delta_{n-1})$. Wold processes are suitable for modeling the dynamics of complex systems, and their inherent Markovian property facilitates the learning task [120, 121]. Similar to the Hawkes process, the interactions among the dimensions of a Wold process can be visualized using a directed graph in which nodes and edges represent processes and direct influences, respectively.

[1] This chapter is based on [45].

Recently, Vaz de Melo et al. [121] showed that Wold processes can model the dynamics of real-world communications more faithfully than the widely used Hawkes processes. Figueiredo et al. [49] then developed a Markov Chain Monte Carlo (MCMC) sampling-based algorithm, called Granger-Busca, to infer the parameters of a Wold process. However, the choice of prior of the MCMC algorithm in [49] requires certain restrictive assumptions on the network. For instance, it requires that every node in the underlying network of a Wold process has at least one out-going edge, *i.e.*, at least one child. Clearly, many practical systems violate this assumption. Beside relaxing the limiting assumptions in [49], we propose an efficient Bayesian algorithm for learning a general class of Wold processes. To achieve scalability, we propose a variational inference (VI) approach to approximate the high-dimensional posterior of the model parameters given the data.

5.2 Related Works

The inference problem for multivariate point processes has been mostly studied for Hawkes processes. There are two types of approaches for estimating the parameters of Hawkes processes. Maximum likelihood-based approaches estimate the parameters from the likelihood of observations [116, 133, 142], whereas cumulant-based approaches learn the parameters of interest by solving a set of equations obtained from various order statistics of the Hawkes process [2, 7, 58]. Some studies address the problem from a Bayesian perspective; *e.g.*, Linderman and Adams [76] develop an MCMC sampling-based algorithm. Due to the long memory of Hawkes processes, the method does not scale well with the number of observations. To improve the scalability of such Bayesian methods, Linderman and Adams [77] propose approximating the continuous-time Hawkes process with a discrete-time formulation. This led to a computationally more efficient stochastic variational inference (VI) algorithm that scales better for longer sequences of observations, at the expense of information loss incurred by binning the events into discrete-time bins. Analogous to [77], the Bayesian approach that we propose here uses mean-field VI to learn the parameters of Wold processes.

Recall that Wold processes are defined through a Markovian transition probability distribution on the inter-event times $p(\delta_{n+1}|\delta_n)$, which measures the probability of the next inter-event time δ_{n+1}, given the preceding one. It turns out that for general Markovian transition probabilities, this model is analytically intractable [56]. However, in the univariate setting, when the transition probabilities have the exponential form $p(\delta_{n+1}|\delta_n) = f(\delta_n)\exp\left(-f(\delta_n)\delta_{n+1}\right)$, the process shows interesting properties [30, 33, 34]. In particular, the next inter-event time δ_{n+1} is then exponentially distributed with rate $f(\delta_n)$. In the case where $f(\delta_n) = \lambda\delta_n^{-1/2}$, the stationary distribution of inter-event times can also be found via Mellin transforms [125]. Similarly, in the case where $f(\delta_n) = \beta+\alpha\delta_n$, the stationary distribution $p(\delta_n)$ has the form $(\beta + \alpha\delta_n)^{-1}\exp\left(-\beta\delta_n\right)$. The analytical properties of a specific type of Wold process that is an infinite process defined on the unit circle are discussed in Isham [60].

Recent efforts consider variations of the exponential Wold process [3, 120]. Instead of defining the Wold process in terms of its inter-event exponential rate, these works define the process in terms of the conditional mean of an exponentially distributed random variable $\mathbb{E}[\delta_{n+1}|\delta_n] = \beta + \alpha\delta_n$. This class of point processes is called a *self-feeding process*. This form of Wold process is able to capture both exponential and power-law behavior, which often occur simultaneously in real data. Realizations of this process tend to generate bursts of intense activity, followed by long periods of silence. Vaz de Melo et al. [121] use self-feeding processes to model the time intervals between communication events for different technologies and means of communications, including short-message services, mobile phone-calls, and e-mail transactions. Building on this work, Figueiredo et al. [49] introduce a multivariate version of the self-feeding process and propose an MCMC sampling-based algorithm to learn the parameters. However, the approach requires restrictive structural assumptions on the network of the process, which limits the applicability of the model.

5.3 Preliminary Definitions

In this section, we describe the model and the notation used throughout this chapter. We first define the univariate Wold process, and then generalize it to the multivariate case.

Consider a temporal point process $\mathcal{T} = \{t_n\}_{n\geq 1}$ on $\mathbb{R}_{\geq 0}$ that index the occurrences of random asynchronous events. Let $\{\delta_n = t_n - t_{n-1}\}_{n\geq 1}$ denote the sequence of inter-event times. As introduced in Example 1.8, $\mathcal{T}$ is called a Wold process if the distribution over the inter-events is Markovian, *i.e.*,

$$p(\delta_{n+1}|\delta_n, \delta_{n-1}, \cdots, \delta_1) = p(\delta_{n+1}|\delta_n). \tag{5.1}$$

The form of the transition probability specifies the class of Wold process. For instance, in this work, we consider the self-feeding process formulation where transition probabilities have the exponential form given by $p(\delta_{n+1}|\delta_n) = f(\delta_n)\exp(-f(\delta_n)\delta_{n+1})$. In addition, we consider $f(\delta_n)$ to be $1/(\beta + \delta_n)$ so that the conditional mean is linear [3, 121].

Now, to define the multivariate case, consider a set of Wold processes $\mathcal{T} = \bigcup_{i=1}^{d} \mathcal{T}_i$ that are observed simultaneously, where $\mathcal{T}_i = \{t_{i,1} < t_{i,2} < \ldots\}$ and $t_{i,n}$ denotes the n-th event in the i-th process. Note that, to make the notation lighter to read in this chapter, we adopt the compact notation $t_{i,n}$, thus indicating both the dimension and the index of the event as a subscript. At a given time t, the conditional intensity of the i-th process depends on the last inter-event times $\{\Delta_{i,j}(t) : j \in [d]\}$ where

$$\Delta_{i,j}(t) \coloneqq s_i(t) - s_j(s_i(t)). \tag{5.2}$$

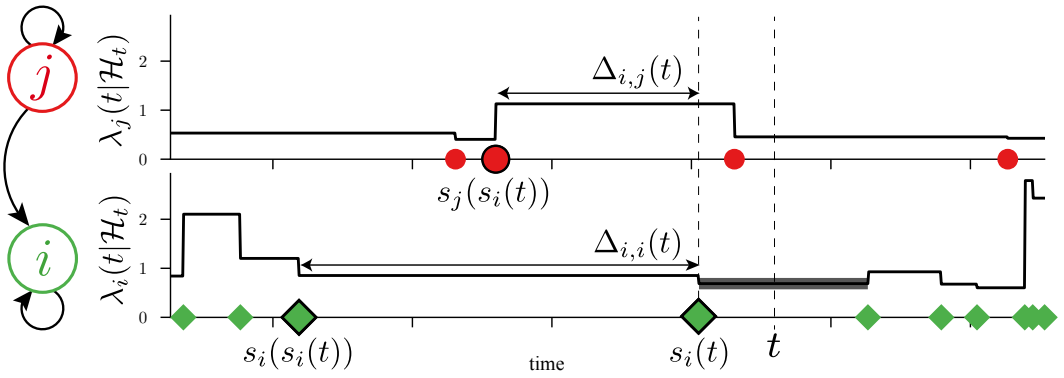


Figure 5.1 – Illustration of the Wold process dynamics on a simple toy example with 2 processes, where process i is influenced by process j and by itself, *i.e.*, $\alpha_{i,j} > 0$ and $\alpha_{i,i} > 0$, and process j also influences itself. At the highlighted time t, the intensity in process i depends on the two highlighted inter-event times $\Delta_{i,j}(t)$ and $\Delta_{i,i}(t)$, which remain constant until the next event in process i.

In this definition, $s_i(t)$ is the last event time of process i before time t, *i.e.*,

$$s_i(t) \coloneqq \max\{t_{i,n} : t_{i,n} < t\}, \tag{5.3}$$

and $s_j(s_i(t))$ is the last event of process j preceding the event $s_i(t)$, *i.e.*,

$$s_j\left(s_i(t)\right) \coloneqq \max\{t_{j,n} : t_{j,n} < s_i(t) < t\}. \tag{5.4}$$

An illustration of the process is shown in Figure 5.1. The conditional intensity function of the i-th process is then

$$\lambda_i(t|\mathcal{H}_t) = \mu_i + \sum_{j=1}^{d} \frac{\alpha_{i,j}}{\beta_{i,j} + \Delta_{i,j}(t)}, \tag{5.5}$$

where $\mu_i \geq 0$ is its background rate, and the influence of process j on process i at time t is captured by $\alpha_{i,j}/(\beta_{i,j} + \Delta_{i,j}(t))$. The parameter $\alpha_{i,j} \geq 0$ is the weight of the influence and $\beta_{i,j} > 0$ ensures the stability of the process, *i.e.*, that the expected number of events stays finite in a finite time horizon [34].

Unlike the Hawkes process, the Wold process has finite memory because of its Markov property. In addition, because $\Delta_{i,j}(t)$ changes only when there is an event in dimension i, a given process i in a Wold process is influenced by other processes (including itself) only when an event occurs in process i, as illustrated in Figure 5.1.

In Hawkes processes, the structure of the Granger causality graph is encoded in the support of the *excitation matrix* [43, 44]. Therefore, learning the support of the excitation matrix is sufficient for recovering the network structure. Analogously, one can gather the

influences among dimensions of a Wold process in a matrix called the *influence matrix* $\mathbf{\Phi}(t) \coloneqq [\alpha_{i,j}/(\beta_{i,j} + \Delta_{i,j}(t))]_{i,j=1}^{d}$. The main reason for this name is that the influence matrix captures the Granger causality of the Wold process. Specifically, the support of $\mathbf{\Phi}(t)$ is the adjacency matrix of the corresponding Granger causality graph.

In this work, we learn the set of parameters

$$\begin{aligned} \boldsymbol{\mu} &\coloneqq \{\mu_i : i \in [d]\}, \\ \boldsymbol{\alpha} &\coloneqq \{\alpha_{i,j} : i, j \in [d]\}, \\ \text{and} \quad \boldsymbol{\beta} &\coloneqq \{\beta_{i,j} : i, j \in [d]\}. \end{aligned}$$

It is worth emphasizing that the algorithm proposed in [49] assumes that $\sum_{i=1}^{d} \alpha_{i,j} = 1$ and $\beta_{i,j} = \beta_i$ for all $i \in [d]$. Herein, we relax all these restrictive assumptions.

5.4 Proposed Learning Approach

5.4.1 Maximum Likelihood Estimation

Suppose that we observe a sequence of discrete events $\mathcal{T} = \bigcup_{i=1}^{d} \mathcal{T}_i$ over an observation period $[0, T]$ generated by a Wold process. The generic approach to infer the parameters of the model is to use regularized maximum-likelihood estimation. The design of regularization depends on the problem at hand, as well as the necessary conditions we are imposing, *e.g.*, positivity or sparsity of the parameters. The log-likelihood function of a multivariate point process can be written as

$$\log p(\mathcal{T}|\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{d} \sum_{t_{i,n} \in \mathcal{T}_i} \log \lambda_i(t_{i,n}|\mathcal{H}_t) - \sum_{i=1}^{d} \int_0^T \lambda_i(t|\mathcal{H}_t) dt. \tag{5.6}$$

The specific form of Wold process defined in (5.5), makes the log-likelihood function non-convex with respect to $\boldsymbol{\beta}$. Moreover, maximum-likelihood estimation of point processes typically scales poorly to high dimensional settings. Therefore, we use a variational inference approach to circumvent both issues of non-convexity and scalability.

5.4.2 Variational Inference

Variational inference (VI) is a method for approximating the posterior distribution over the model parameters given the observations. In order to represent the posterior in a tractable form, it is common to define an auxiliary variable that relates the parameters and the observations [49, 77, 113]. Observing that the conditional intensity in (5.5) is a summation of $d + 1$ terms, we can use the superposition theorem of point processes to define the parent of each event [34, 76]. More precisely, we define an auxiliary variable

$\boldsymbol{z}_{i,n}$ for each event $t_{i,n}$ to be a one-hot vector that indicates the cause of that event. This cause is either the background rate μ_i or one of the processes in $\{1, \ldots, d\}$. Specifically,

$$\boldsymbol{z}_{i,n} = \left[z_{i,n}^{(0)}, z_{i,n}^{(1)}, \cdots, z_{i,n}^{(d)}\right].$$

where $z_{i,n}^{(0)}$ is 1 if and only if $t_{i,n}$ was caused by the background rate μ_i or $z_{i,n}^{(j)}$ is 1 if and only if $t_{i,n}$ was caused by process j. Because an event has exactly one cause (or parent), $\boldsymbol{z}_{i,n}$ is a one-hot vector, which means that $\sum_{j=0}^{d} z_{i,n}^{(j)} = 1$ for all i and n.

Our approach is conceptually similar to the VI algorithm proposed in [77] for learning Hawkes processes. However, because Hawkes processes suffer from long memory, each preceding event is a potential parent, so the number of auxiliary variables increases exponentially with the number of events. To overcome this issue, Linderman and Adams [77] approximate the Hawkes process by discretizing time, which has the drawback of introducing an approximation error. In contrast, as a result of the Markovian nature of the Wold process, only the preceding events of each dimension are the potential parents. Thus, the number of potential parents of an event remains constant.

Having defined the auxiliary variable $\boldsymbol{z}$, we approximate the posterior distribution $p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta} | \mathcal{T})$ with a variational distribution $q(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ that minimizes the KL-divergence between p and q. In particular, VI solves for the optimal variational distribution that minimizes the KL-divergence, or equivalently it maximizes the evidence lower bound (ELBO), given by

$$\text{ELBO}(q) = \mathbb{E}_q \left[\log p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathcal{T})\right] - \mathbb{E}_q \left[\log q(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta})\right]. \tag{5.7}$$

We consider a *mean-field approximation* for the variational distribution. In such an approximation, the variational parameters are assumed to be independent. Therefore,

$$q(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_{i=1}^{d} q(\mu_i) \times \prod_{i=1}^{d} \prod_{n=1}^{|\mathcal{T}_k|} q(\boldsymbol{z}_{i,n}) \times \prod_{i=1}^{d} \prod_{j=1}^{d} q(\alpha_{i,j})\, q(\beta_{i,j}). \tag{5.8}$$

Using this approximation and coordinate ascent for maximizing (5.7), we obtain the variational distributions $\{q(\mu_i), q(\boldsymbol{z}_{i,n}), q(\alpha_{i,j}), q(\beta_{i,j})\}$ by selecting appropriate prior distributions over the parameters. Coordinate ascent is a commonly used optimization method in VI. It iteratively updates each factor of the mean-field variational density while holding the others unchanged [123]. Next, we give the variational updates. Derivation of these updates can be found in Appendix A.4.1.

Variational update of the auxiliary parent variable $\boldsymbol{z}_{i,n}$. The definition of the auxiliary variable $\boldsymbol{z}_{i,n}$ implies that $\sum_{j=0}^{d} z_{i,n}^{(j)} = 1$. As shown in Appendix A.4.1, this results in

$$q(\boldsymbol{z}_{i,n}) = \text{Categorical}\left(d+1\ ;\ p_{i,n}^{(0)}, ..., p_{i,n}^{(d)}\right), \tag{5.9}$$

where the probabilities

$$p_{i,n}^{(0)} \propto \exp\left(\mathbb{E}_{q(\mu_i)}[\log \mu_i]\right) \tag{5.10}$$

$$\text{and} \quad p_{i,n}^{(j)} \propto \exp\left(\mathbb{E}_{q(\alpha_{i,j})}[\log(\alpha_{i,j})] - \mathbb{E}_{q(\beta_{i,j})}[\log\left(\beta_{i,j} + \Delta_{i,j}(t_{i,n})\right)]\right), \forall j \in [d] \tag{5.11}$$

are normalized such that $\sum_{j=0}^{d} p_{i,n}^{(j)} = 1$.

Variational update of $\alpha_{i,j}$. Selecting a Gamma distribution with shape $a_{i,j}$ and rate $b_{i,j}$ for prior of $\alpha_{i,j}$ results in a Gamma mean-field approximation of the posterior, given by

$$q(\alpha_{i,j}) = \text{Gamma}\left(A_{i,j}; B_{i,j}\right), \tag{5.12}$$

where

$$A_{i,j} \coloneqq a_{i,j} + \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{q(z_{i,n}^{(j)})}\left[z_{i,n}^{(j)}\right],$$

$$B_{i,j} \coloneqq b_{i,j} + \sum_{i=1}^{|\mathcal{T}_i|} \mathbb{E}_{q(\beta_{i,j})}\left[\frac{t_{i,n} - t_{i,n-1}}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right].$$

Variational update of μ_i. Similar to α, we use a Gamma distribution with shape c_i and rate d_i as the prior of μ_i resulting in the posterior, given by

$$q(\mu_i) = \text{Gamma}\left(C_i; D_i\right), \tag{5.13}$$

where

$$C_i \coloneqq c_i + \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{q(z_{i,n}^{(0)})}\left[z_{i,n}^{(0)}\right],$$

$$D_i \coloneqq d_i + \sum_{n=1}^{|\mathcal{T}_i|} t_{i,n} - t_{i,n-1}.$$

Variational update of $\beta_{i,j}$. For this parameter, we select the prior distribution to be Inverse-Gamma with shape $\phi_{i,j}$ and scale $\psi_{i,j}$. This choice of prior results in a variational distribution of $\beta_{i,j}$ proportional to

$$(\beta_{i,j})^{-\phi_{i,j}-1} e^{-\frac{\psi_{i,j}}{\beta_{i,j}}} \prod_{n=1}^{|\mathcal{T}_i|} \left[(\beta_{i,j} + \Delta_{i,j}(t_{i,n}))^{-\mathbb{E}\left[z_{i,n}^{(j)}\right]} \exp\left(- \frac{\mathbb{E}[\alpha_{i,j}](t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})} \right) \right]. \quad (5.14)$$

This form of the density function is not straightforward to work with as we need to compute $\mathbb{E}[\log(\beta_{i,j} + \Delta_{i,j}(t))]$ and $\mathbb{E}[1/(\beta_{i,j} + \Delta_{i,j}(t))]$. However, the form of this distribution suggests that it can be well-approximated by an inverse-Gamma distribution. Hence, we approximate this distribution with an Inverse-Gamma and use the following variational update

$$q(\beta_{i,j}) = \text{Inverse-Gamma}(\Phi_{i,j}; \Psi_{i,j}), \quad (5.15)$$

where $\Phi_{i,j}$ and $\Psi_{i,j}$ are selected so that its moments coincide with the moments of the distribution in (5.14). This leads to the following form of the parameters

$$\Phi_{i,j} := \frac{w x_w - v x_v}{x_w - x_v} - 1,$$
$$\Psi_{i,j} := \frac{(w - v) x_v x_w}{x_w - x_v}.$$

In these equations, $w \geq 1$, $w > v > 0$ and x_w denotes the smallest positive real root of equation $g_w(x) = 0$, where

$$g_w(x) := \frac{\phi_{i,j} + 1 - w}{x} + \sum_{n=1}^{|\mathcal{T}_i|} \frac{\mathbb{E}_{q(z_{i,n}^{(j)})}[z_{i,n}^{(j)}]}{x + \Delta_{i,j}(t_{i,n})} - \frac{\psi_{i,j}}{x^2} - \sum_{n=1}^{|\mathcal{T}_i|} \frac{\mathbb{E}_{q(\alpha_{i,j})}[\alpha_{i,j}](t_{i,n} - t_{i,n-1})}{(x + \Delta_{i,j}(t_{i,n}))^2}. \quad (5.16)$$

The following lemma guarantees the existence of such an inverse-Gamma distribution.

Lemma 5.1. *If $0 < \phi_{i,j} + 1 - w < \phi_{i,j} + 1 - v$ and $w \geq 1$, then $\Phi_{i,j}$ and $\Psi_{i,j}$ exist and are positive.*

Proof. Let $g_u(x)$ denote the function in (5.16). We have $\lim_{x \to 0_+} g_u(x) = -\infty$ and $\lim_{x \to \infty} g_u(x) = 0_+$ for $u = v, w$. Thus, $g_u(x)$ has at least one positive real root. Without loss of generality, let x_v and x_w be the smallest positive real roots of $g_v(x)$ and $g_w(x)$, respectively. Given the assumption in the lemma, it is clear that $g_v(x) > g_w(x)$ for $x > 0$. Hence, $0 = g_v(x_v) > g_w(x_v)$ and $g_v(x_w) > g_w(x_w) = 0$. Since x_w is the smallest positive

root of $g_w(x)$, $\lim_{x\to 0_+} g_u(x) = -\infty$, and $g_w(x_v) < 0$, then $x_w > x_v$. Now, using the facts that $w > v$ and $w \geq 1$, and the equations of $\Phi_{i,j}$ and $\Psi_{i,j}$, we conclude the proof. □

In Section 5.5.3, we provide an example of realizations of the distribution in (5.14) to illustrate the goodness of this approximation.

5.5 Experimental Results

We now provide a comparison of our VI approach with state-of-the-art approaches in two sets of experiments. We first simulate synthetic realizations of Wold processes, where the ground-truth parameters are known, to measure the performance and efficiency of each approach to recover the influence matrix. Subsequently, we evaluate our approach on two real-world datasets of multivariate asynchronous time series. For reproducibility, we provide a detailed description of the setup of each experiment in Appendix C.4 and make the code publicly available online at `https://github.com/trouleau/var-wold`.

5.5.1 Experiments on Synthetic Data

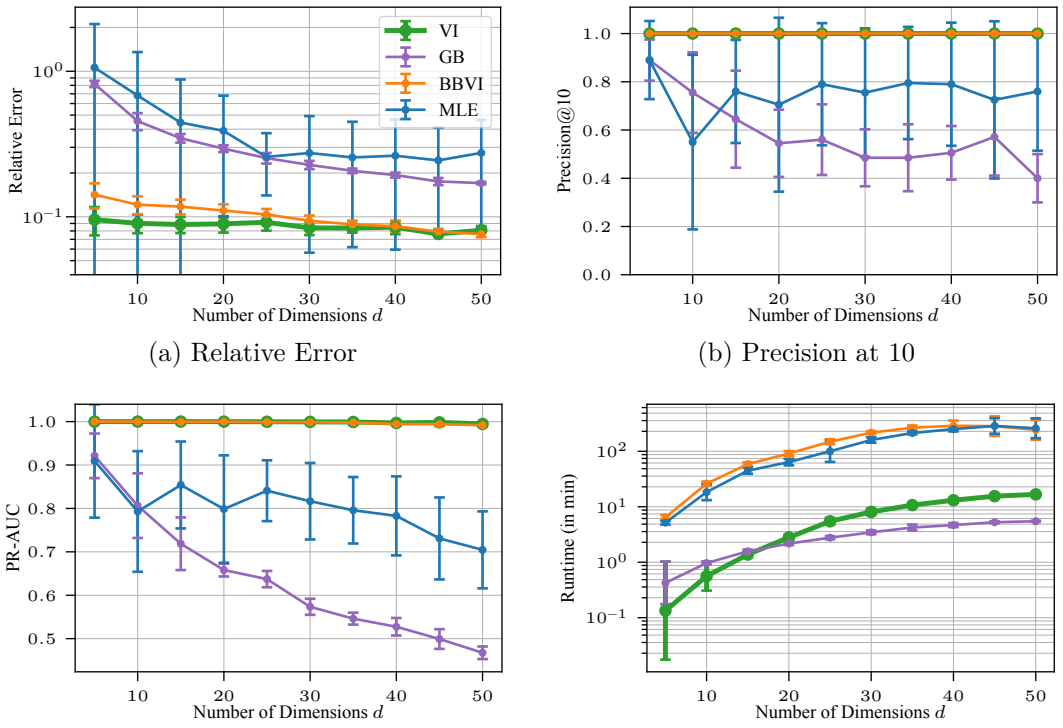
To simulate Wold processes, we generated Erdős–Rényi random graphs with d nodes. We sampled background rates $\{\mu_i^*\}$ from Uniform$[0, 0.05]$, edge weights $\{\alpha_{i,j}^*\}$ from Uniform$[0.1, 0.2]$ for all edges, and parameters $\{\beta_{i,j}^*\}$ from Uniform$[1, 2]$, all independently. To evaluate the scalability of an approach with respect to the number of dimensions, we varied the number of dimensions d between 5 and 50 nodes. The results are averaged over 5 graphs with 4 realizations of the Wold process for each graph, with an average of 10 000 training events per dimension. We compared the performance of our approach, denoted as VI, with three other methods:

GB. The MCMC sampling-based approach Granger-Busca from [49] is the only other approach designed for Wold processes. Note that GB does not estimate a posterior for $\{\beta_{i,j}\}$, but instead uses the data-driven heuristic $\beta_{i,j} = \text{median}(\{t_{i,n+1} - t_{i,n} | t_{i,n} \in \mathcal{T}_i\})/\exp(1)$, referred to as *Busca*, as advised by the authors.

BBVI. To compare with another method based on VI, we adapted the approach discussed in Chapter 2, for learning Wold processes. The approach is based on black-box VI and the variational EM algorithm.

MLE. For a simple baseline, we also compared with maximum-likelihood estimation with a Tikhonov regularizer.

Note that the three Bayesian approaches VI, BBVI, and GB estimate a posterior over the parameters rather than a point-estimate as done in MLE. Therefore, we use the mean of the posteriors to evaluate the performance of the estimated influence weights $\{\hat{\alpha}_{i,j}\}$.



(a) Relative Error

(b) Precision at 10

(c) PR-AUC

(d) Runtime

Figure 5.2 – Results on synthetic data for varying number of dimensions K. Panels (a) (log-scale) relative error, (b) precision@10, (c) PR-AUC and panel (d) (log-scale) empirical runtime of each approach in minutes.

Similar to the previous chapters, we evaluate the performance of each approach in learning the influence matrix of the processes. More precisely, we compared the estimated $\{\hat{\alpha}_{i,j}\}$ with the ground-truth $\{\alpha^*_{i,j}\}$ using three metrics common in the literature:

- **Relative error.** To evaluate the distance of the estimated weights to the ground-truth ones, we computed the averaged relative error defined as $|\hat{\alpha}_{i,j} - \alpha^*_{i,j}|/\alpha^*_{i,j}$ when $\alpha^*_{i,j} \neq 0$, and $\hat{\alpha}_{i,j}/(\min_{\alpha^*_{n,m}>0} \alpha^*_{n,m})$ when $\alpha^*_{i,j} = 0$ [130, 107].

- **Precision@k.** To assess the performance of the approaches at recovering the top edges, we used precision@k, which is defined as the average fraction of correctly identified edges in the top k largest estimated weights [49].

- **PR-AUC.** Considering that an edge exists in the influence matrix if the learned value $\hat{\alpha}_{i,j} > \eta$, we evaluate the performance of the resulting binary edge classification problem using the area under the precision-recall curve over all thresholds $\eta > 0$.

Our results are depicted in Figure 5.2. As shown in Figure 5.2a-5.2c, both VI and BBVI outperform GB and MLE on all metrics. Despite the non-convexity of the problem,

both methods achieve an almost perfect precision@10 and PR-AUC for all numbers of dimensions. On the other hand, MLE showed a large variance in the estimated parameters[2].

The computational complexity per iteration is of order $\mathcal{O}(|\mathcal{T}|d)$ for our VI algorithm and $\mathcal{O}(|\mathcal{T}|\log d)$ for GB, where $|\mathcal{T}|$ is the total number of events and d is the number of dimensions. However, note that each update of our VI approach is easily parallelizable over each of the d^2 edges, while GB can only be parallelized over each of the d nodes. We also observed that VI empirically requires fewer iterations to converge compared to GB. To show this, we compared the runtime of each method in Figure 5.2d. Note that all methods were implemented in Python, GB was compiled in Cython, and VI used just-in-time compilation with the library Numba. To make runtime comparison fair, all methods were run on a single core on the same machine. Although both VI and BBVI perform well, the runtime of VI is about one order of magnitude faster than BBVI and is similar to GB. More details are available in Appendix B.3.

5.5.2 Experiments on Real Datasets

We evaluated the approaches on two datasets from the Snap Network Repository[3]: (1) the email-Eu-core dataset that contains emails sent between collaborators from a large European research institution [49, 98], and (2) the Memetracker dataset containing online blog posts [2, 49, 74]. We compare our VI approach on these datasets with GB, which currently is the most scalable approach. In [49], the authors showed that Wold processes are better suited than Hawkes processes for these two datasets.

Email-EU-core. The dataset consists of source nodes (senders) that send events to destination nodes (receivers) at some time. Each event is represented as a triplet (source, destination, timestamp). Following the same preprocessing as [49], we aggregated the events by receiver and considered the top 100 receivers, *i.e.*, those with the most events, resulting in a total of 92 924 events. We hypothesize that the ground-truth influence matrix is determined by the number of emails sent by a sender to a receiver. More precisely, the ground-truth was defined as a graph whose nodes are both senders and receivers and whose directed edges captured the flow of communication from sender to receiver, weighted by fraction of received emails. We used the first 75% of the dataset for training and the remaining 25% for testing. We evaluated the results for two tasks: (1) An edge-estimation task where we evaluated the performance of each approach to recover the ground-truth influence matrix of the training set, and (2) an event-prediction task where we measured the predictive log-likelihood of the two approaches on the held-out test set. Because both approaches estimate the posterior distribution over the parameters

[2]To highlight that the discrepancy of performance does not come from the particular experimental setup, we present additional results in Appendix B.3 for an alternative experimental setup matching the structural assumption of GB, *i.e.*, where $\sum_i \alpha^*_{i,j} = 1$.

[3]https://snap.stanford.edu/data/

(in contrast with single point-estimators), we characterized the uncertainty using Monte Carlo samples of the parameters from the learned posterior distributions, and we reported the mean and standard deviations among these samples for all the reported metrics.

The results on the Email-EU-core dataset are shown in Table 5.1. VI outperfoms GB on all metrics. The improvement can be explained by the fact that VI relaxes the restrictive assumptions in GB, *i.e.*, that $\sum_{i=1}^{d} \alpha_{i,j} = 1$ and $\beta_{i,j} = \beta_i \ \forall i \in [d]$, which we discussed in Section 5.3.

Table 5.1 – Results on the EU-email-core dataset.

	PR-AUC	Precision@10	Precision@50	Precision@200	Pred. log-likelihood
VI	**0.33** (±0.00)	**0.40** (±0.00)	**0.40** (±0.00)	**0.47** (±0.00)	**−5.64** (±1.16e−2)
GB	0.32 (±0.00)	0.20 (±0.00)	0.35 (±0.07)	0.43 (±0.03)	−11.56 (±1.78e−2)

MemeTracker. The dataset consists of the times of publication of online blog posts along with the hyperlinks within. The dataset was originally collected to analyze the propagation of short phrases, called *memes*, and is often modeled as a multivariate point process [2, 49, 106]. To evaluate the performance of their algorithms, [2] and [49] extracted a ground-truth influence matrix based on hyperlink references among the websites and reported the precision of their methods, which were low. This could be explained by the presence of noise in the dataset[4], as well as by non-stationarity, (*i.e.*, varying dynamics of the data over time), which were reported by Rodriguez et al. [106]. Therefore, for the MemeTracker dataset, we focused on the predictive capability of our algorithm compared to GB by evaluating the predictive log-likelihood on held-out data. More precisely, we split the data into observation windows of about 12 days, trained on each window and tested on the following one. The log-likelihood values were normalized by number of events[5].

Figure 5.3 depicts the results on the MemeTracker dataset. Again, to account for uncertainty in the estimation, we also reported the mean and standard deviations of the predictive log-likelihood among Monte Carlo samples of the parameters. We see that VI outperforms GB for all observation windows. Moreover, the values are not stable over time, confirming the findings of Rodriguez et al. [106] that the dynamics of the data are indeed non-stationary.

[4]The assumption that a hyperlink (source) appearing in another blog (destination) implies a causal influence might not be accurate. For example, a hyperlink can appear in comments of a blog, unrelated from its main content.

[5]For reproducibility, we provide the detailed preprocessing steps in Appendix C.4.2.

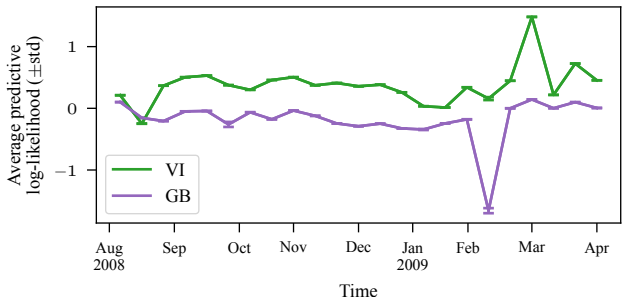


Figure 5.3 – Held-out predictive log-likelihood on the MemeTracker dataset.

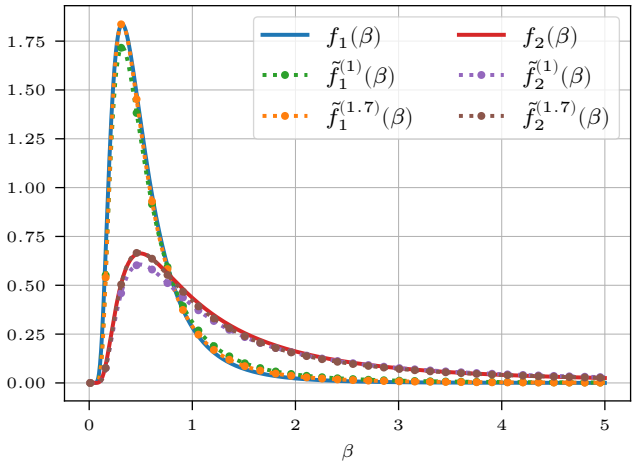


Figure 5.4 – Two examples of the distribution in (5.14) and their corresponding Inverse-Gamma approximation in (5.15). The Inverse-Gamma are denoted by tilde and they are obtained by selecting $v = 0$ and $w \in \{1, 1.7\}$.

5.5.3 Example of the $q(\beta_{i,j})$ Approximation

Next, we evaluate the goodness of the approximation proposed in (5.15) for the update of $\beta_{i,j}$. To do so, we considered two realizations of the distribution in (5.14), given by

$$f_1(\beta) \propto \beta^{-1-1} \cdot e^{-1/\beta} \cdot (\beta + 2.9)^{-0.2} \cdot e^{-\frac{0.6}{\beta+2.9}} \cdot (\beta + 1.7)^{-0.8} \cdot e^{\frac{-1.6}{\beta+1.7}}, \tag{5.17}$$

$$f_2(\beta) \propto \beta^{-3-1} \cdot e^{-1/\beta} \cdot (\beta + 0.3)^{-0.3} \cdot e^{\frac{-1.6}{\beta+0.3}} \cdot (\beta + 1.1)^{-1.8} \cdot e^{\frac{-0.4}{\beta+1.1}} \cdot (\beta + 2)^{-0.1} \cdot e^{\frac{-.8}{\beta+2}}, \tag{5.18}$$

and we computed their approximated Inverse-Gamma distribution using (5.15). We display the resulting distributions in Figure 5.4. The approximated Inverse-Gamma distributions are denoted by $\widetilde{f}$ and are obtained by selecting $v = 0$ and $w \in \{1, 1.7\}$.

In order to measure the goodness of the approximation, we also present in Table 5.2 the KL-divergence between (5.17) and (5.18) and their approximated Inverse-Gamma distributions for several choices of w.

Table 5.2 – KL-divergences between the distributions in (5.17) and (5.18) and their approximations.

	$\widetilde{f}^{(1)}$	$\widetilde{f}^{(1.3)}$	$\widetilde{f}^{(1.7)}$	$\widetilde{f}^{(1.9)}$	$\widetilde{f}^{(2.5)}$
f_1	0.0370	0.0272	0.0126	0.0070	–
f_2	0.0296	0.0222	0.0151	0.0119	0.0062

5.6 Summary

We have addressed the problem of learning the Granger causality graph of multivariate temporal point processes. This problem has been widely studied for the multivariate Hawkes process, but the long memory of such processes makes Bayesian inference difficult. Because of its Markovian intensity function, the Wold process does not suffer from the same shortcomings and has therefore recently gained popularity in the literature. We relaxed the limiting structural assumptions of the only available state-of-the-art method and proposed an efficient Bayesian algorithm based on variational inference for the multivariate Wold process with exponential transition probabilities. Our experiments on both synthetic and real-world datasets show that our approach outperforms the state-of-the art and is able to accurately and efficiently recover the influence matrix of the process.

6 Conclusion and Outlook

In this thesis, we have considered the challenges faced when modeling event data with self-exciting temporal point processes. We have addressed, in particular, the statistical and algorithmic aspects of uncovering the diffusion patterns of interdependent events from observed data, with a focus on the widely used Hawkes process. We have explored the impact of several types of noise present in event data, characterized the sensitivity of common learning methods to these sources of noise, and have proposed solutions to handle them. We have studied each setting through controlled experiments in both synthetic and real-world datasets.

- In Chapter 2, we have explored how to learn the excitation matrix of a Hawkes process when limited amounts of data are available for training. In this context, overfitting becomes a major issue and regularization is of paramount importance. We have exploited recent advances in variational inference and have proposed an expectation-maximization algorithm that enables us to use advanced regularization schemes and automatically learn an extended set of hyperparameters. This approach is also able to learn a versatile class of posterior distribution over the parameters of the Hawkes process.
- In Chapter 3, we have addressed the setting where the time of events cannot be recorded accurately, thus leading to a synchronization noise between different types of events. We have characterized the robustness of the maximum likelihood estimator to synchronization noise and have demonstrated that even a small amount of noise can lead to a significant bias in the estimated parameters. We have shown that this noise makes the likelihood function non-smooth and introduces discontinuities that makes the optimization task particularly challenging. We have proposed an algorithm that overcomes these challenges and accurately recovers the parameters of the Hawkes process for a wide range of noise values.
- In Chapter 4, we have considered a more general type of temporal noise, called random translations, where the timestamps of events are subject to random and

unknown shifts that are independently drawn from some unknown probability distribution. We have taken a more theoretical approach and have proved that the cumulants of the Hawkes process are invariant to random translations. We have validated our findings empirically, by showing that cumulant-based estimation methods can robustly learn the parameters of the process from randomly translated events, whereas maximum-likelihood based methods are brittle.

- Finally, in Chapter 5, we have considered another type of temporal point process, called the Wold process; it answers a computational limitation of the Bayesian treatment of Hawkes processes and has been shown to be well-suited to model real-world communication dynamics. We have relaxed the restrictive assumptions made in the stats-of-the-art Wold process models and have proposed a scalable Bayesian approach based on variational inference for inferring the parameters of the process.

The approach we have taken in this thesis consists of establishing the shortcomings of common learning algorithms for temporal point processes, in order to enable reliable insights for policy makers. However, there remains important research questions that have yet to be explored. We discuss promising research directions along which our work could be extended, as well as the main challenges that need to be addressed.

Learning Hawkes Processes under Quantized Data. Whether it is due to privacy concerns or limitations of the data-collection procedure, the time of events might be quantized in discrete-time bins. With the growing popularity of point process models, recent studies have applied such aggregated event data to temporal point processes. For example, both Chen et al. [26] and Chiang et al. [28] developed temporal point process models and used data released publicly by The New York Times on daily COVID-19 cases in the state of New Jersey. Similarly, Mohler et al. [89] and Bertozzi et al. [18] used an extension of the Hawkes processes to compute the effective reproduction number of early stages of the COVID-19 epidemic in China, based on the number of deaths per day. However, temporal point process models are designed in continuous time and assume that two events cannot occur simultaneously[1]. This raises the following questions: Can temporal point processes accurately capture the relationships between event types when only quantized events are available? And if so, what classes of algorithms are better suited for quantized events? Even though this setting has not directly been studied for Hawkes processes, discretization of time were introduced in a few studies, as a way to develop a scalable inference algorithm [67, 68, 77].

[1] We discuss this assumption, known as the property of *simple* point processes, in Equation (1.5) of Section 1.2.

On Neural-Based Point Processes. Neural-based point processes enable to capture more complex conditional intensity functions that the linear dependencies of Hawkes processes. However, this complexity is usually achieved at the expense of interpretability of the model. Although the influence structure of Hawkes (or Wold) processes can be conveniently summarized into a Granger causality graph, this is not the case of most neural-based models. At the time of writing, only the recent models from Xiao et al. [129] and from Zhang et al. [140] can provide summary statistics for Granger causality that is enabled by their use of an attention mechanism. Learning interpretable dynamics for complex processes is certainly one of the most promising research directions for enabling actionable insights to be understood and trusted by policy makers. Nevertheless, due to their complexity, neural-based point processes are usually simply fit using maximum likelihood estimation. We have seen in Chapters 3 and 4 that the likelihood function of Hawkes processes is sensitive to noise in the observed timestamps. Because most neural-based models are inspired by Hawkes processes and encode the history of the process into a vector representation, it is therefore critical to evaluate the impact of noisy observations on these kinds of models.

On the Need for Better Data Collection. Designing better point processes and understanding the shortcomings of their inference algorithms addresses only the consequences of a larger problem. Collecting better data tackles the problem at its source. As such, developing digital technologies that enable easier, faster, and more reliable collection of event data is certainly of even greater importance than studying the algorithmic aspects of noisy data.

In conclusion, the results we have presented in this thesis highlight the often neglected challenges of modeling event data with point processes. Our hope is that this discussion will spark the interest of researchers to address these issues in future studies. As several research problems remain open, they offer an opportunity to further study the challenges of modeling event data with self-exciting temporal point processes.

A Technical Details

A.1 Technical Details of Chapter 2

A.1.1 Simple Optimization over Hyperparameters in MAP Estimation

In this section, we show that we cannot simply find $\boldsymbol{\alpha}$ by optimizing the negative log-likelihood in (2.3) or the MAP objective in (2.7) over $\boldsymbol{\alpha}$.

First note that, minimizing regularized negative log-likelihood in (2.3) over $\boldsymbol{\alpha}$, simply sets $\boldsymbol{\alpha}$ to infinity. Second, we show that maximizing the MAP objective in (2.7) over $\boldsymbol{\alpha}$ also fails because it is unbounded from above. We show this for the case of the Gaussian prior defined by

$$p_{\boldsymbol{\alpha}}(\boldsymbol{\mu}, \boldsymbol{W}) = p_{\boldsymbol{\alpha}_\mu}(\boldsymbol{\mu}) p_{\boldsymbol{\alpha}_W}(\boldsymbol{W}) = \frac{1}{\sqrt{2\pi\boldsymbol{\alpha}_\mu}} \exp\left(-\frac{\|\boldsymbol{\mu}\|^2}{2\boldsymbol{\alpha}_\mu}\right) \cdot \frac{1}{\sqrt{2\pi\boldsymbol{\alpha}_W}} \exp\left(-\frac{\|\boldsymbol{W}\|^2}{2\boldsymbol{\alpha}_W}\right). \tag{A.1}$$

but the same result holds for other priors. The log of the Gaussian prior (A.1) is

$$\begin{aligned} \log p_{\boldsymbol{\alpha}}(\boldsymbol{\mu}, \boldsymbol{W}) &= \log p_{\boldsymbol{\alpha}_\mu}(\boldsymbol{\mu}) + \log p_{\boldsymbol{\alpha}_W}(\boldsymbol{W}) \\ &= -\frac{\|\boldsymbol{\mu}\|^2}{2\boldsymbol{\alpha}_\mu} - \frac{\|\boldsymbol{W}\|^2}{2\boldsymbol{\alpha}_W} - \frac{1}{2}\log \boldsymbol{\alpha}_\mu - \frac{1}{2}\log \boldsymbol{\alpha}_W + c, \end{aligned} \tag{A.2}$$

where c is a constant independent of $\boldsymbol{\alpha}$. In the MAP objective (2.7), if we set $\boldsymbol{\mu} = 1$ and $\boldsymbol{W} = 0$, *i.e.*, all processes are simple Poisson process with rate 1 and no interaction between them, then the conditional intensity $\lambda_i(t) = 1$ for all $i \in [d]$ and $t \geq 0$. The log-likelihood in (2.4) becomes $\log p(\mathcal{S}|\boldsymbol{\mu}, \boldsymbol{W}) = -DT$, which is bounded from below. Set $\boldsymbol{\alpha}_\mu = 1$, then for $\boldsymbol{\alpha}_W \to 0^+$, we get $\log p_{\boldsymbol{\alpha}}(\boldsymbol{\mu}, \boldsymbol{W}) \to \infty$. Hence, the MAP estimator for $\boldsymbol{\alpha}$ is unbounded from above and maximizing the MAP objective simultaneously over both the hyperparameters $\boldsymbol{\alpha}$ and the model parameters $\boldsymbol{\mu}$ and $\boldsymbol{W}$ would fail.

A.2 Technical Details of Chapter 3

A.2.1 Derivation of the Gradient

We derive the gradient of the log-likelihood of the DESYNC-MHP model with respect to the synchronization noise parameters $\boldsymbol{z}$. It corresponds to the update rule of $\boldsymbol{z}$ in Algorithm 3.1. First, the gradient of (3.3) with respect to the noise parameter in the k-th dimension can be written as

$$\nabla_{z_k} \log p(\widetilde{\mathcal{S}}|\boldsymbol{z}, \theta) = \frac{\partial}{\partial z_k} \left[\sum_{i=1}^{d} \left(\sum_{\tau \in \widetilde{N}_i(T)} \log \lambda_i(\tau - z_i | \widetilde{\mathcal{H}}_{\tau - z_i}) - \int_{t_0 - z_i}^{T - z_i} \lambda_i(t|\widetilde{\mathcal{H}}_t) dt \right) \right],$$

where

$$\lambda_i(t|\widetilde{\mathcal{H}}_t) = \mu_i + \sum_{j=1}^{d} \sum_{\tau \in \widetilde{\mathcal{H}}_t^j} \phi_{i,j}(t - \tau) = \mu_i + \sum_{j=1}^{d} \int_0^t \tilde{\phi}_{i,j}(t - \tau) d\widetilde{N}_j(\tau).$$

Substituting the above intensity into the log-likelihood implies

$$\begin{aligned} \frac{\partial}{\partial z_k} \sum_{j=1}^{d} \Bigg\{ & \sum_{\tau \in \widetilde{\mathcal{H}}_T^j} \log \left(\mu_j + \sum_{i=1}^{d} \sum_{s \in \widetilde{\mathcal{H}}_\tau^i} w_{j,i} \tilde{\phi}(\tau - z_j - s + z_i) \right) \\ & - \mu_j (T' - t_0') - \sum_{i=1}^{d} \int_{t_0'}^{T'} \int_0^{T' - s + z_i} w_{j,i} \tilde{\phi}(t) dt d\widetilde{N}_i(s) \Bigg\}, \end{aligned} \tag{A.3}$$

where $t_0' \coloneqq t_0 - \min_i z_i$, $T' \coloneqq T - \max_i z_i$, and

$$\tilde{\phi}(t) = \frac{e^{-\beta t} + e^{-(\gamma - \beta')t}}{1 + e^{-\gamma t}}.$$

is the smooth approximation of the exponential kernel defined in Equation (3.5). Note that in the above equation, we approximated the boundary of the integral by $[t_0', T')$ to account for windowing effects.

First, we compute the derivative of $\tilde{\phi}(\tau - z_k - s + z_i)$ with respect to z_k,

$$\begin{aligned} \frac{\partial}{\partial z_k} \tilde{\phi}(u_{k,i}) = & - \frac{(e^{-\beta u_{k,i}} + e^{-(\gamma - \beta')u_{k,i}}) \gamma e^{-\gamma u_{k,i}}}{(1 + e^{-\gamma u_{k,i}})^2} \\ & + \frac{\left(\beta e^{-\beta u_{k,i}} + (\gamma - \beta') e^{-(\gamma - \beta')u_{k,i}} \right)}{1 + e^{-\gamma u_{k,i}}}, \end{aligned} \tag{A.4}$$

where $u_{k,i} \coloneqq \tau - z_k - s + z_i$. Further, $\frac{\partial}{\partial z_i}\tilde{\phi}(u_{k,i}) = -\frac{\partial}{\partial z_k}\tilde{\phi}(u_{k,i})$. The other important term is

$$\frac{\partial}{\partial z_k}\int_{t_0'}^{T'-s+z_k}\tilde{\phi}(t)dt = \tilde{\phi}(T'-s+z_k). \tag{A.5}$$

By taking the derivative of the log-function, Equation (A.3) can be written as follows

$$\begin{aligned}\sum_{j=1}^{d}\Bigg\{\sum_{\tau\in\widetilde{\mathcal{H}}_T^j}\frac{\sum_{i=1}^{d}\sum_{s\in\widetilde{\mathcal{H}}_\tau^i}w_{j,i}\nabla_{z_k}\tilde{\phi}(\tau-z_j-s+z_i)}{\mu_j+\sum_{i=1}^{d}\sum_{s\in\widetilde{\mathcal{H}}_\tau^i}w_{j,i}\tilde{\phi}(\tau-z_j-s+z_i)}\\ -\sum_{i=1}^{d}\int_{t_0'}^{T'}\left(\nabla_{z_k}\int_0^{T'-s+z_i}w_{j,i}\tilde{\phi}(t)dt\right)d\widetilde{N}_i(s)\Bigg\},\end{aligned} \tag{A.6}$$

Substituting (A.4) and (A.5) into (A.6) implies the result.

The gradient of the log-likelihood with respect to $w_{k,l}$ for some k and l in $\{1,\dots,d\}$ is

$$\sum_{\tau\in\widetilde{\mathcal{H}}_T^k}\frac{\sum_{s\in\widetilde{\mathcal{H}}_\tau^l}\tilde{\phi}(\tau-z_k-s+z_l)}{\mu_k+\sum_{s\in\widetilde{\mathcal{H}}_\tau^b}w_{k,l}\tilde{\phi}(\tau-z_k-s+z_l)}-\int_{t_0'}^{T'}\int_0^{T'-s+z_l}\tilde{\phi}(t)dtd\widetilde{N}_l(s). \tag{A.7}$$

In (A.7), we have

$$\int_0^x\tilde{\phi}(t)dt \approx \frac{1-e^{-\beta x}}{\beta}+\frac{\log 2-\log(1+e^{-(\gamma-\beta')x})}{\gamma-\beta'},$$

when $\gamma \gg \beta$ and $\gamma/(\gamma-\beta') \approx 1$.

A.3 Technical Details of Chapter 4

A.3.1 Proof of Theorem 4.1

We first start by showing that a random translation of an inhomogeneous Poisson process is again a Poisson process [34, 39]. Subsequently, we will use this result to compute the cumulants of a randomly-translated Hawkes process.

Lemma A.1. *(Section 2.3 of Daley and Vere-Jones [34]) Let $N(\cdot)$ be an inhomogeneous Poisson process on $\mathbb{R}_{\geq 0}$ with intensity $\lambda(t)$. The resulting process after a random translation with distribution function $F(\cdot)$ is yet another Poisson process with intensity*

$$\widetilde{\lambda}(t) \coloneqq \int_{-\infty}^{t}\lambda(t-x)F(dx). \tag{A.8}$$

Proof. We prove this lemma by showing that the number of events within an arbitrary interval of the randomly-translated Poisson process is Poisson distributed with the specified rate. Let $I_\theta = [0, \theta]$ be an interval in $\mathbb{R}$, then

$$\mathbb{P}(N(I_\theta) = k) = \frac{(\Lambda(I_\theta))^k}{k!} e^{-\Lambda(I_\theta)},$$

where $\Lambda(I_\theta)$ is $\int_0^\theta \lambda(x)dx$. The probability that an arrival at time $t \in I_\theta$ is translated to an arbitrary interval $I \coloneqq [a, b]$ is

$$\mathbb{P}(a \leq x + t \leq b) = F(b - t) - F(a - t) \coloneqq F(I - t)$$

The probability that only m of k arrivals within I_θ are translated to I is equal to

$$\binom{k}{m} \left(\int_0^\theta F(I - t)\lambda(t)dt / \Lambda(I_\theta) \right)^m \left(\int_0^\theta (1 - F(I - t))\lambda(t)dt / \Lambda(I_\theta) \right)^{k-m}.$$

Therefore, the probability that m events are observed in I after translation is

$$\begin{aligned}
&\lim_{\theta\to\infty} \sum_{k\geq m} \frac{(\Lambda(I_\theta))^k}{k!} e^{-\Lambda(I_\theta)} \binom{k}{m} \left(\int_0^\theta F(I - t)\lambda(t)dt / \Lambda(I_\theta) \right)^m \\
&\qquad \cdot \left(\int_0^\theta (1 - F(I - t))\lambda(t)dt / \Lambda(I_\theta) \right)^{k-m} \\
&= \lim_{\theta\to\infty} \sum_{k\geq m} \frac{e^{-\Lambda(I_\theta)}}{m!(k-m)!} \left(\int_0^\theta F(I - t)\lambda(t)dt \right)^m \cdot \left(\int_0^\theta (1 - F(I - t))\lambda(t)dt \right)^{k-m} \\
&= \lim_{\theta\to\infty} \frac{e^{-\Lambda(I_\theta)}}{m!} \left(\int_0^\theta F(I - t)\lambda(t)dt \right)^m \sum_{k\geq 0} \frac{1}{k!} \left(\int_0^\theta (1 - F(I - t))\lambda(t)dt \right)^k \\
&= \lim_{\theta\to\infty} \frac{e^{-\Lambda(I_\theta)}}{m!} \left(\widetilde{\Lambda}(I_\theta) \right)^m e^{\left(\Lambda(I_\theta) - \widetilde{\Lambda}(I_\theta)\right)},
\end{aligned}$$

where $\widetilde{\Lambda}(I_\theta) \coloneqq \int_0^\theta F(I - t)\lambda(t)dt$. Letting θ to go to infinity and setting $I = \{v, v + dv\}$, we obtain

$$\begin{aligned}
&\mathbb{P}(d\tilde{N}(v) = 1) = \widetilde{\lambda}(v)dv = \left(\int_0^\infty f(v - u)\lambda(u)du \right) dv \\
&= \left(\int_{-\infty}^v \lambda(v - x)F(dx) \right) dv,
\end{aligned}$$

where $F(dv) = f(v)dv$. □

To establish the result in Theorem 4.1, it is also useful to prove the following lemma.

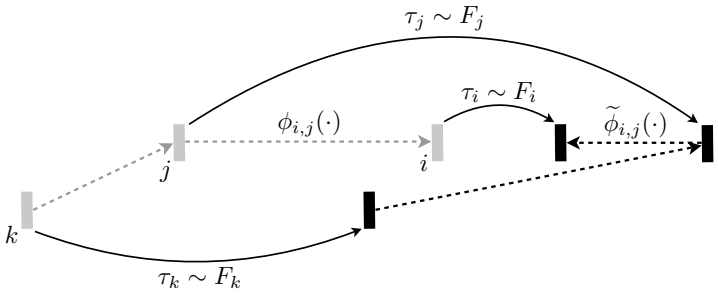


Figure A.1 – The evolution of a part of a cluster after random translation. Nodes are labeled by their types.

Lemma A.2. *For every multi-index* $\boldsymbol{i} = (i_1, ..., i_n)$ *and every vector* $\tilde{\boldsymbol{t}} = (\tilde{t}_1, ..., \tilde{t}_n)$*, we have*

$$K_{\boldsymbol{i}}(\tilde{\boldsymbol{t}})d\tilde{\boldsymbol{t}} = \mathbb{P}(E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}} \cap C^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}),$$

where $E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}$ *denotes the event that for every* k *there is a type* i_k *events at time* $\tilde{t}_k$ *and* $C^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}$ *is the event that there exists a cluster such that all the events in* $\tilde{\boldsymbol{t}}$ *belong to that cluster.*

Proof. This can be seen from the fact that

$$\mathbb{E}[dN_{i_1}(\tilde{t}_1)...dN_{i_n}(\tilde{t}_n)] = \mathbb{P}\left(\forall k \in \{1, ..., n\}, \text{ there is a type } i_k \text{ event at } \tilde{t}_k\right) = \mathbb{P}(E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}})$$

and

$$\mathbb{P}(E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}) = \mathbb{P}(E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}} \cap C^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}) + \mathbb{P}(E^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}} \cap \bar{C}^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}),$$

where $\bar{C}^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}$ denotes the complement of the event $C^{\boldsymbol{i}}_{\tilde{\boldsymbol{t}}}$. The rest follows similarly to the proof of Equation (24) in Appendix A of [62]. □

Lemma A.3. *Consider the setting in Theorem 4.1, and define*

$$\widetilde{R}_{i,j}(t)dt \coloneqq \mathbb{P}\left(\textit{type } j \textit{ event at 0 causes type } i \textit{ event at } t\right), \tag{A.9}$$

then $\widetilde{R}_{i,j}(t) = \left[\sum_{n\geq 0} \widetilde{\boldsymbol{\Phi}}^{*n}(t)\right]_{i,j}$.

Proof. Suppose in a cluster C of the original Hawkes process, *i.e.*, before translation, dimension j at some time y triggers an arrival in dimension i. Based on the definition of the clusters, the arrival times in dimension i are distributed as an inhomogeneous Poisson process with rate $\phi_{i,j}(t-y)$. This evolution of this cluster is illustrated in Figure A.1.

Suppose that nodes j and i are translated by τ_j and τ_i, respectively. Then $\tilde{t}_j = y + \tau_j$ and $\tilde{t}_i = y + x + \tau_i$, where $x \sim \text{Exponential}(\phi_{i,j}(t-y))$, and $\tilde{t}_i - \tilde{t}_j = x + \tau_i - \tau_j$.

The above observation and Lemma A.1 imply that

$$\frac{\mathbb{P}(dN_i(\tilde{t}_i) = 1|\tau_j)}{dt} = \int_{\mathbb{R}} f_i(\tilde{t}_i - s)\phi_{i,j}(s - y)ds = \int_{\mathbb{R}} f_i(\tilde{t}_i - \tilde{t}_j + \tau_j - s)\phi_{i,j}(s)ds,$$

where N_i denotes the number of arrivals in i-th dimension of the translated process. Therefore,

$$\widetilde{\phi}_{i,j}(\tilde{t}_i - \tilde{t}_j) \coloneqq \frac{\mathbb{P}(dN_i(\tilde{t}_i) = 1)}{dt} = \int_{\mathbb{R}} \int_{\mathbb{R}} f_i(\tilde{t}_i - \tilde{t}_j + \tau_j - s)\phi_{i,j}(s)f_j(\tau_j)dsd\tau_j.$$

This equation may be interpreted as follows: the cluster of a Hawkes process after the random translation forms a new cluster in which dimension j at some time $\tilde{t}_j$ causes an offspring of type i by generating a realization of an inhomogeneous Poisson process with rate $\widetilde{\phi}_{i,j}(t - \tilde{t}_j)$. Moreover, an immigrant from dimension k appears in the translated cluster with rate $\int_{\mathbb{R}} \mu_k F_k(dx) = \mu_k$.

Define $p^n_{i,j}(t)$ as the probability that an event of type j at 0, after n generations, causes a type i event at t in the translated cluster. Clearly, $p^0_{i,j}(t) = [\boldsymbol{I}\delta(t)]_{i,j}dt$. For $n > 0$, we have

$$\begin{aligned}
p^1_{i,j}(t) &= \widetilde{\phi}_{i,j}(t)dt, \\
p^2_{i,j}(t) &= \sum_{k=1}^{d} \int_{\mathbb{R}} \widetilde{\phi}_{k,j}(s)\widetilde{\phi}_{i,k}(t - s)dsdt = [\widetilde{\boldsymbol{\Phi}}^{*2}(t)]_{i,j}dt, \\
&\vdots \\
p^n_{i,j}(t) &= [\widetilde{\boldsymbol{\Phi}}^{*n}(t)]_{i,j}dt.
\end{aligned}$$

This implies

$$\widetilde{R}_{i,j}(t)dt = \sum_{n\geq 0} p^n_{i,j}(t) = \left[\sum_{n\geq 0} \widetilde{\boldsymbol{\Phi}}^{*n}(t)\right]_{i,j} dt.$$

□

With Lemma A.3 at hand, we are ready to prove Equations (4.3)-(4.5) of Theorem 4.1.

First, for Equation (4.3), the definition of the cumulant leads to

$$K_i(\tilde{t})d\tilde{t} = K(dN_i(\tilde{t})) = \mathbb{E}[dN_i(\tilde{t})].$$

The last term in the above expression is the probability that an immigrant, say from dimension j, generates an event in dimension i at time $\tilde{t}$, which equals

$$\sum_{j=1}^{d} \int_{\mathbb{R}} \mu_j \widetilde{R}_{i,j}(\tilde{t}-x)dx = \sum_{j=1}^{d} \int_{\mathbb{R}} \mu_j \widetilde{R}_{i,j}(x)dx = K_i.$$

Then, for Equation (4.4), we have

$$K_{i,j}(\tilde{t}_1,\tilde{t}_2)d\tilde{t}_1 d\tilde{t}_2 = \mathbb{P}\Big(\text{types } i,j \text{ events at times } \tilde{t}_1,\tilde{t}_2 \text{ within the same cluster}\Big).$$

The above event happens if and only if node i and j have a common ancestor in a cluster, which happens with probability

$$\begin{aligned}
&\sum_{k=1}^{d} \int_{\mathbb{R}} \mathbb{P}\Big(\text{an immigrant generates an event in dimension } k \text{ at time } x\Big) \\
&\times \mathbb{P}\Big(k \text{ generates events in dimensions } i \text{ and } j \text{ at times } \tilde{t}_1 \text{ and } \tilde{t}_2\Big) d\tilde{t}_1 d\tilde{t}_2 dx \\
&= \sum_{k=1}^{d} \int_{\mathbb{R}} K_k \times \Big(\widetilde{R}_{i,k}(\tilde{t}_1 - x)\widetilde{R}_{j,k}(\tilde{t}_2 - x)d\tilde{t}_1 d\tilde{t}_2\Big)dx
\end{aligned}$$

Finally, for Equation (4.5), we use the above Lemmas A.2 and A.3, and the fact that i, j and k can all occur in one cluster if one of the followings cases happen:

- $\{i,j\}$ and $\{k\}$,
- $\{i,k\}$ and $\{j\}$,
- $\{k,j\}$ and $\{i\}$,
- $\{i,j,k\}$,

where types within one set have a common ancestor and separate sets have a common ancestor. For example, in case of $\{i,j\}$ and $\{k\}$, i and j share a common ancestor and the common ancestor of $\{i,j\}$ and $\{k\}$ have their own common ancestor. Say the common ancestor of i and j be from type m at some time y, and assume m and k have a different common ancestor than $\{i,j\}$, say n at another time x. This case can be formally written as follows

$$\begin{aligned}
&\text{case}(\{i,j\},\{k\}) \\
&= \sum_{m,n=1}^{d} \int_{\mathbb{R}} K_n \widetilde{R}_{k,n}(\tilde{t}_3 - x)d\tilde{t}_3 \int_{\mathbb{R}} \widetilde{R}_{i,m}(\tilde{t}_1 - y)d\tilde{t}_1 \widetilde{R}_{j,m}(\tilde{t}_2 - y)d\tilde{t}_2 \widetilde{\Psi}_{m,n}(y-x)dydx \\
&= \sum_{m=1,n=1}^{d} K_n \int_{\mathbb{R}}\int_{\mathbb{R}} \widetilde{R}_{k,n}(\tilde{t}_3 - x)\widetilde{R}_{i,m}(\tilde{t}_1 - y)\widetilde{R}_{j,m}(\tilde{t}_2 - y)\widetilde{\Psi}_{m,n}(y-x)dydxd\tilde{t}_1 d\tilde{t}_2 d\tilde{t}_3.
\end{aligned}$$

In this expression, $\widetilde{\Psi}_{m,n}$ appears because nodes m and n cannot coincide, then there must be at least one generation difference from n to m. The probability of this event is

$$\begin{aligned}\sum_{r>0} p^r_{m,n}(y-x) &= \left[\sum_{r>0} \widetilde{\mathbf{\Phi}}^{*r}(y-x)\right]_{m,n} \\ &= \left[\widetilde{\mathbf{R}}(y-x) - \boldsymbol{I}\delta(y-x)\right]_{m,n} \\ &\coloneqq \widetilde{\Psi}_{m,n}(y-x).\end{aligned}$$

Similarly, one can compute the probability of the other partitions and conclude the result.

A.3.2 Proof of Corollary 4.2

Recall from Theorem 4.1 that $\widetilde{\phi}_{i,j}(t) = f_i * \phi_{i,j} * \underline{f}_j(t)$. Since functions $\phi_{i,j}$, f_i, and f_j are bounded, their convolutions is also bounded. We have

$$\overline{\phi}_{i,j} = \int_{\mathbb{R}}\int_{\mathbb{R}}\int_{\mathbb{R}} f_i(t+x-s)\phi_{i,j}(s)f_j(x)dsdxdt = \int_{\mathbb{R}} \phi_{i,j}(s)ds.$$

The last equality is due to the fact that $\int_{\mathbb{R}} f_i(t)dt = \int_{\mathbb{R}} f_j(t)dt = 1$. By assumption, the matrix $\mathcal{L}[\mathbf{\Phi}](0)$ has spectral radius less than one, and by the above equality so does the matrix $\overline{\mathbf{\Phi}}$.

A.3.3 Proof of Corollary 4.3

The result is immediate from the definition of the covariance density matrix of a randomly-translated Hawkes process,

$$\widetilde{\Sigma}_{i,j}(\tilde{t}_1, \tilde{t}_2) \coloneqq K_{i,j}(\tilde{t}_1, \tilde{t}_2) - \frac{\mathbb{E}[dN_i(\tilde{t}_1)]}{d\tilde{t}_1}\epsilon_{i,j}\delta(\tilde{t}_1 - \tilde{t}_2),$$

and substituting the second order cumulant density given in (4.4).

A.3.4 Estimators for the Integrated Cumulants

We used the same empirical estimates for the cumulants as in [2] as follows.

$$\begin{aligned}\widehat{K}_i &= \frac{1}{T}\sum_{\tau\in\tilde{\mathbf{t}}_i} 1 = \frac{N_i(T)}{T}, \\ \widehat{\overline{K}}_{i,j} &= \frac{1}{T}\sum_{\tau\in\tilde{\mathbf{t}}_i} \left(N_j(\tau+H) - N_j(\tau-H) - 2H\widehat{K}_j\right),\end{aligned}$$

$$\widehat{\overline{K}}_{i,j,k} = \frac{1}{T}\sum_{\tau\in\tilde{\mathbf{t}}_i}\big(N_j(\tau+H)-N_j(\tau-H)-2H\widehat{K}_j\big)\big(N_k(\tau+H)-N_k(\tau-H)-2H\widehat{K}_k\big) - \frac{\widehat{\overline{K}}_i}{T}\sum_{\tau\in\tilde{\mathbf{t}}_j}\sum_{\tau'\in\tilde{\mathbf{t}}_k}\max\{(2H-|\tau'-\tau|),0\}+4H^2\widehat{K}_i\widehat{K}_j\widehat{K}_k,$$

where $\tilde{\mathbf{t}}_i$ denotes the set of all events up to time T in dimension i and H is a hyper-parameter used to truncate the interval $(-\infty,\infty)$ to $[-H,H]$. See [2] for a proof of the consistency of these estimators.

A.3.5 Discussion on the Covariance Density Matrix Equations

In this section we show that equations (4.12) and (4.14) do not admit unique solutions. First we need the following Lemma.

Lemma A.4. *For a stationary Hawkes process and a random translation of it, both* $\rho(\mathcal{L}[\mathbf{\Phi}](s))$ *and* $\rho(\mathcal{L}[\widetilde{\mathbf{\Phi}}](s))$ *are strictly less than one for all* $s\in\mathbb{C}$.

Proof. From Gelfand's Formula, we know that for any matrix $\mathbf{B}$, $\rho(\mathbf{B})=\lim_{n\to\infty}||\mathbf{B}^n||^{1/n}$, where $||\cdot||$ is any matrix norm. We will apply this formula with $\mathbf{B}=\mathcal{L}[\mathbf{\Phi}](s)$ and $||\cdot||$ chosen as the max norm $||\cdot||_{max}$, but first observe

$$\begin{aligned}\|\mathcal{L}[\mathbf{\Phi}]^n(s)\|_{max} &\coloneqq \max_{i,j}\left|\left[\mathcal{L}[\mathbf{\Phi}]^n(s)\right]_{i,j}\right| \\ &\leq \max_{i,j}\left|\left[\mathcal{L}[\mathbf{\Phi}]^n(0)\right]_{i,j}\right| = \|\mathbf{\Phi}^n\|_{max},\ \forall n\in\mathbb{N}, s\in\mathbb{C}. \quad \text{(A.10)}\end{aligned}$$

We used the triangle inequality and the fact that for a positive function f,

$$|\mathcal{L}[f](s)|\leq|\mathcal{L}[f](0)|.$$

Now, applying Gelfand's Formula, we obtain

$$\rho\left(\mathcal{L}[\mathbf{\Phi}](s)\right)\leq\lim_{n\to\infty}||\mathbf{\Phi}^n||_{max}^{1/n}=\rho\left(\mathbf{\Phi}\right)<1.$$

The last inequality is due to the stationarity assumption of the Hawkes process. Following the same steps and the result of Corollary 4.2, one can show $\rho(\mathcal{L}[\widetilde{\mathbf{\Phi}}](s))<1$. □

Lemma A.5. *Equations* (4.12) *and* (4.14) *do not admit unique solutions in terms of* $\mathbf{\Phi}(t)$ *and* $\widetilde{\mathbf{\Phi}}(t)$*, respectively.*

Proof. We only present the proof for equation (4.12). The argument for (4.14) is similar. Let $\mathbf{\Phi}(t)$ denote a solution to (4.12) and $\mathbf{R}(t)\coloneqq\boldsymbol{I}\delta(t)+\mathbf{\Psi}(t)$, then

$$\mathbf{\Sigma}(t)=\underline{\mathbf{R}}*\mathbf{\Lambda}\mathbf{R}^T(t)-\mathbf{\Lambda}\delta(t).$$

Define $\mathbf{R}_0(t) \coloneqq \mathbf{R}(t)\mathbf{\Lambda}^{1/2}\mathbf{O}\mathbf{\Lambda}^{-1/2}$, where $\mathbf{O}$ is any orthogonal matrix, *i.e.*, any matrix such that $\mathbf{O}\mathbf{O}^T = \mathbf{O}^T\mathbf{O} = \mathbf{I}$. It is easy to see that

$$\mathbf{\Sigma}(t) = \underline{\mathbf{R}}_0 * \mathbf{\Lambda}\mathbf{R}_0^T(t) - \mathbf{\Lambda}\delta(t).$$

Lemma A.4 implies that $\mathcal{L}[\mathbf{R}](s)$ is bounded and equals $(\mathbf{I} - \mathcal{L}[\mathbf{\Phi}](s))^{-1}$, therefore

$$\begin{aligned}
\mathcal{L}[\mathbf{R}_0](s) &= \mathcal{L}[\mathbf{R}](s)\mathbf{\Lambda}^{1/2}\mathbf{O}\mathbf{\Lambda}^{-1/2} \\
&= (\mathbf{I} - \mathcal{L}[\mathbf{\Phi}](s))^{-1}\mathbf{\Lambda}^{1/2}\mathbf{O}\mathbf{\Lambda}^{-1/2} \\
&= \Big(\mathbf{\Lambda}^{1/2}\mathbf{O}^T\mathbf{\Lambda}^{-1/2}(\mathbf{I} - \mathcal{L}[\mathbf{\Phi}](s))\Big)^{-1} \\
&= \Big(\mathbf{I} - \mathbf{I} + \Lambda^{1/2}\mathbf{O}^T\mathbf{\Lambda}^{-1/2} - \mathbf{\Lambda}^{1/2}\mathbf{O}^T\mathbf{\Lambda}^{-1/2}\mathcal{L}[\mathbf{\Phi}](s)\Big)^{-1} \\
&= \Big(\mathbf{I} - \mathbf{A}(s)\Big)^{-1},
\end{aligned}$$

where

$$\mathbf{A}(s) \coloneqq \mathbf{I} - \mathbf{\Lambda}^{1/2}\mathbf{O}^T\mathbf{\Lambda}^{-1/2} + \mathbf{\Lambda}^{1/2}\mathbf{O}^T\mathbf{\Lambda}^{-1/2}\mathcal{L}[\mathbf{\Phi}](s).$$

This means that $\mathbf{\Phi}_0(t) = \mathcal{L}^{-1}[\mathbf{A}](t)$ is also a solution of (4.12).

□

A.4 Technical Details of Chapter 5

A.4.1 Derivations of the Variational Inference Updates

In this section, we present the derivations of variational updates for the Wold process parameters. From [20], we know that maximizing the ELBO with the mean-field assumption implies that the variational update of a parameter x_j from the parameter set $\boldsymbol{x}$ given the observation set $\boldsymbol{d}$ has the following form

$$q(x_j) = \exp\Big(\mathbb{E}_{-x_j}\left[\log p(\boldsymbol{x}, \boldsymbol{d})\right]\Big) + \text{const.} \tag{A.11}$$

In the above expression, $p(\boldsymbol{x}, \boldsymbol{d})$ denotes the joint distribution of the parameters and the observations. The expectation is taken with respect to the variational density of all the parameters except x_j. Using this update rule, we can explicitly derive all the variational updates of interest. For notational simplicity, we use the following notations throughout the appendix.

$$\begin{aligned}
&\boldsymbol{\alpha}_i \coloneqq \{\alpha_{i,j}\}_{j=1}^d, \quad \boldsymbol{\alpha} \coloneqq \{\boldsymbol{\alpha}_i\}_{i=1}^d, \\
&\boldsymbol{\beta}_i \coloneqq \{\beta_{i,j}\}_{j=1}^d, \quad \boldsymbol{\beta} \coloneqq \{\boldsymbol{\beta}_i\}_{i=1}^d, \\
&\boldsymbol{z} \coloneqq \{\boldsymbol{z}_{i,n} : n \in [|\mathcal{T}_i|]\}_{i=1}^d, \quad \boldsymbol{\mu} \coloneqq \{\mu_i\}_{i=1}^d.
\end{aligned}$$

Variational update for the auxiliary parent variables $\boldsymbol{z}_{i,n}$

Let $-\boldsymbol{z}_{i,n}$ denote the set of all parameters except $\boldsymbol{z}_{i,n}$. From (A.11), we obtain

$$\begin{aligned}\log\left(q\left(\boldsymbol{z}_{i,n}\right)\right) &= \mathbb{E}_{-\boldsymbol{z}_{i,n}}\left[\log p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathcal{T})\right] + \text{const.} \\ &= \mathbb{E}_{-\boldsymbol{z}_{i,n}}\left[\log p\left(\boldsymbol{z}_{i,n}|\mu_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \mathcal{T}\right)\right] + \text{const.}\end{aligned}$$

The last equality holds because of the mean-field assumption. In order to obtain the conditional distribution of the parent variable given the rest of the parameters, we use the fact that the number of events in a given interval is distributed according to Poisson distribution. Hence,

$$\begin{aligned}p\left(\boldsymbol{z}_{i,n}|\mu_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \mathcal{T}\right) =& \text{Poisson}\left(z_{i,n}^{(0)}; \mu_i(t_{i,n} - t_{i,n-1})\right) \\ &\times \prod_{j=1}^{d} \text{Poisson}\left(z_{i,n}^{(j)}; \frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right) \mathbb{1}_{\left\{\sum_{j=0}^{d} z_{i,n}^{(j)} = 1\right\}},\end{aligned} \tag{A.12}$$

where $\mathbb{1}_{\{.\}}$ denotes the indicator function. The product form in (A.12) results again the mean-field assumption, and the indicator enforces that $\sum_{j=0}^{d} z_{i,n}^{(j)} = 1$. Substituting the above conditional distribution into the variational update equation, we obtain

$$\begin{aligned}\log\left(q\left(\boldsymbol{z}_{i,n}\right)\right) =& \mathbb{E}_{\mu_i}\left[\log\left(\mu_i(t_{i,n} - t_{i,n-1})\right)^{z_{i,n}^{(0)}}\right] \\ &+ \mathbb{E}_{\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i}\left[\log \prod_{j=1}^{d}\left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right)^{z_{i,n}^{(j)}}\right] \\ &+ \log \mathbb{1}_{\left\{\sum_j z_{i,n}^{(j)} = 1\right\}} + \text{const.} \\ =& z_{i,n}^{(0)}\, \mathbb{E}_{\mu_i}\left[\log\left(\mu_i(t_{i,n} - t_{i,n-1})\right)\right] \\ &+ \sum_{j=0}^{K} z_{i,n}^{(j)}\, \mathbb{E}_{\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i}\left[\log\left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right)\right] \\ &+ \log \mathbb{1}_{\left\{\sum_j z_{i,n}^{(j)} = 1\right\}} + \text{const.}\end{aligned}$$

Therefore, $q(\boldsymbol{z}_{i,n})$ is Categorical, *i.e.*,

$$q(\boldsymbol{z}_{i,n}) = \text{Categorical}\left(d+1; p_{i,n}^{(0)}, ..., p_{i,n}^{(d)}\right), \tag{A.13}$$

where $p_{i,n}^{(j)}$ is the probability that $z_{i,n}^{(j)}$ is one and the others are zero. Therefore, $\left\{p_{i,n}^{(j)}\right\}$ is a valid probability distribution, *i.e.*, $\sum_{j=0}^{d} p_{i,n}^{(j)} = 1$.

Variational update for $\alpha_{i,j}$

From (A.11), we have

$$\begin{aligned}
\log\left(q\left(\alpha_{i,j}\right)\right) &= \mathbb{E}_{-\alpha_{i,j}}\left[\log p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathcal{T})\right] + \text{const.} \\
&= \mathbb{E}_{-\alpha_{i,j}}\left[\log p(\boldsymbol{z}|\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathcal{T}) + \log p(\boldsymbol{\alpha}|\mathcal{T})\right] + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{-\alpha_{i,j}}\left[\log p\left(\boldsymbol{z}_{i,n}\middle|\mu_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \mathcal{T}\right)\right] + \log p(\alpha_{i,j}) + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{-\alpha_{i,j}}\left[z_{i,n}^{(j)}\ \log\left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right) - \left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right)\right] \\
&\quad + \log p(\alpha_{i,j}) + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{z_{i,n}^{(j)}}\left[z_{i,n}^{(j)}\right] \log(\alpha_{i,j}) - \alpha_{i,j} \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{\beta_{i,j}}\left[\frac{t_{i,n} - t_{i,n-1}}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right] \\
&\quad + \log p(\alpha_{i,j}) + \text{const.}
\end{aligned}$$

If we select the prior distribution of $\alpha_{i,j}$ to be Gamma with shape $a_{i,j}$ and rate $b_{i,j}$, the variational posterior remains Gamma, *i.e.*,

$$q(\alpha_{i,j}) = \text{Gamma}\left(A_{i,j}; B_{i,j}\right), \tag{A.14}$$

where the shape and rate parameters are respectively given by

$$A_{i,j} \coloneqq a_{i,j} + \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{z_{i,n}^{(j)}}\left[z_{i,n}^{(j)}\right], \qquad B_{i,j} \coloneqq b_{i,j} + \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{\beta_{i,j}}\left[\frac{t_{i,n} - t_{i,n-1}}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right].$$

Variational update for μ_i

The update rule for μ_i is similar to the one of $\alpha_{i,j}$.

$$\begin{aligned}
\log\left(q\left(\mu_i\right)\right) &= \mathbb{E}_{-\mu_i}\left[\log p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathcal{T})\right] + \text{const.} \\
&= \mathbb{E}_{-\mu_i}\left[\log p(\boldsymbol{z}|\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathcal{T}) + \log p(\boldsymbol{\mu}|\mathcal{T})\right] + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{-\mu_i}\left[\log p\left(\boldsymbol{z}_{i,n}\middle|\mu_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \mathcal{T}\right)\right] + \log p(\mu_i) + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{-\mu_i}\left[z_{i,n}^{(0)}\ \log\left(\mu_i(t_{i,n} - t_{i,n-1})\right) - \mu_i(t_{i,n} - t_{i,n-1})\right] \\
&\quad + \log p(\mu_i) + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{z_{i,n}^{(0)}}\left[z_{i,n}^{(0)}\right] \log(\mu_i) - \mu_i \sum_{n=1}^{|\mathcal{T}_i|} (t_{i,n} - t_{i,n-1}) + \log p(\mu_i) + \text{const.}
\end{aligned}$$

Selecting a Gamma prior with shape c_i and rate d_i implies the result.

Variational update for $\beta_{i,j}$

Note that $\beta_{i,j}$ is defined for i, j in $[d]$. Similar to the update rule for $\alpha_{i,j}$, we have

$$\begin{aligned}
\log\left(q\left(\beta_{i,j}\right)\right) &= \mathbb{E}_{-\beta_{i,j}}\left[\log p(\boldsymbol{\mu}, \boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}|\mathcal{T})\right] + \text{const.} \\
&= \mathbb{E}_{-\beta_{i,j}}\left[\log p(\boldsymbol{z}|\boldsymbol{\mu}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathcal{T}) + \log p(\boldsymbol{\beta}|\mathcal{T})\right] + \text{const.} \\
&= \sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}_{-\beta_{i,j}}\left[z_{i,n}^{(j)} \log\left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right) - \left(\frac{\alpha_{i,j}(t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right)\right] \\
&\quad + \log p(\beta_{i,j}) + \text{const.} \\
&= -\sum_{n=1}^{|\mathcal{T}_i|} \mathbb{E}\left[z_{i,n}^{(j)}\right] \log(\beta_{i,j} + \Delta_{i,j}(t_{i,n})) - \mathbb{E}[\alpha_{i,j}] \sum_{n=1}^{|\mathcal{T}_i|} \frac{t_{i,n} - t_{i,n-1}}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})} \\
&\quad + \log p(\beta_{i,j}) + \text{const.}
\end{aligned}$$

If we select an Inverse-Gamma prior for $\beta_{i,j}$ with shape $\phi_{i,j}$ and scale $\psi_{i,j}$, $q(\beta_{i,j})$ will be proportional to

$$\beta_{i,j}^{-\phi_{i,j}-1} e^{-\frac{\psi_{i,j}}{\beta_{i,j}}} \prod_{n=1}^{|\mathcal{T}_i|} (\beta_{i,j} + \Delta_{i,j}(t_{i,n}))^{-\mathbb{E}\left[z_{i,n}^{(j)}\right]} \exp\left(-\frac{\mathbb{E}[\alpha_{i,j}](t_{i,n} - t_{i,n-1})}{\beta_{i,j} + \Delta_{i,j}(t_{i,n})}\right), \tag{A.15}$$

for $i, j \in [d]$. This distribution is not analytically tractable, but it can be approximated well by an inverse-Gamma distribution. Therefore, we approximate the variational update for $\beta_{i,j}$ as an Inverse-Gamma$(\Phi_{i,j}, \Psi_{i,j})$. We choose its parameters $\Phi_{i,j}$ and $\Psi_{i,j}$ such that its resulting moments coincide with the moments of the distribution in (A.15). Finding the moments of the distribution in (A.15) tends to be quite challenging. Instead, we use the following observation to obtain our approximation.

Remark A.6. *Let $f(x; a, b)$ be the p.d.f. of the Inverse-Gamma distribution with shape a and rate b. The Function $x^u f(x; a, b)$ has a global maximum that occurs at $b/(a+1-u)$ for $u \in \mathbb{R}_+$.*

We argue that if the u-th moment of an Inverse-Gamma variable, with shape $\Phi_{i,j}$ and rate $\Psi_{i,j}$, coincides with the u-th moment of the distribution in (A.15), denoted by $h(x)$, then we should have

$$\int_{\mathbb{R}_+} x^u f(x; \Phi_{i,j}, \Psi_{i,j}) dx = \int_{\mathbb{R}_+} x^u h(x) dx.$$

A sufficient condition for the above equality is that the points that maximize

$$x^u f(x; \Phi_{i,j}, \Psi_{i,j}) \text{ and } x^u h(x)$$

should coincide. This happens if

$$\frac{\Psi_{i,j}}{\Phi_{i,j}+1-u} = x_u, \tag{A.16}$$

where x_u is the point that maximizes $x^u h(x)$. By equating the derivative of $\log(x^u h(x))$ to zero, it is easy to see that x_u is the real root of the following equation

$$\frac{\phi_{i,j}+1-u}{x} + \sum_{n=1}^{|\mathcal{T}_i|} \frac{\mathbb{E}\left[z_{i,n}^{(j)}\right]}{x+\Delta_{i,j}(t_{i,n})} - \frac{\psi_{i,j}}{x^2} - \sum_{n=1}^{|\mathcal{T}_i|} \frac{\mathbb{E}[\alpha_{i,j}](t_{i,n}-t_{i,n-1})}{(x+\Delta_{i,j}(t_{i,n}))^2} = 0.$$

Since the above function has continuous derivatives, we can use, for example, Halley's method to find its root. Equation (A.16) alone cannot specify both $\Psi_{i,j}$ and $\Phi_{i,j}$. Thus, by selecting two different u, say $u = v$ and $u = w$, we obtain

$$\frac{\Psi_{i,j}}{\Phi_{i,j}+1-v} = x_v, \quad \frac{\Psi_{i,j}}{\Phi_{i,j}+1-w} = x_w.$$

Solving for $\Psi_{i,j}$ and $\Phi_{i,j}$, we obtain

$$\Phi_{i,j} = \frac{wx_w - vx_v}{x_w - x_v} - 1, \quad \Psi_{i,j} = \frac{(w-v)x_w x_v}{x_w - x_v}.$$

Lemma 5.1 implies that such x_v and x_w exist and the above shape and scale are positive for appropriate choices of v, w, and $\phi_{i,j}$.

A.4.2 Computing the required statistics

Note that the variational updates introduced in Section A.4.1 depend on each others through some common statistics. For instance, the variational update for the auxiliary variable $\boldsymbol{z}_{i,n}$ in (A.13) requires computing $\mathbb{E}_{\alpha_{i,j}}[\log \alpha_{i,j}]$. In this section, we provide analytical expressions of such statistics.

Since $q(\boldsymbol{z}_{i,n})$ is Categorical, for $i \in [d]$ and $n \in \mathcal{T}_i$, we have

$$\mathbb{E}_{z_{i,n}^{(j)}}\left[z_{i,n}^{(j)}\right] = p_{i,n}^{(j)}, \text{ for } j \in [d] \cup \{0\}, \tag{A.17}$$

where $p_{i,n}^{(j)}$ is the probability that $z_{i,n}^{(j)} = 1$ and $z_{i,n}^{(k)} = 0$ for $k \neq j$.

Given that $\alpha_{i,j}$ has a Gamma $(A_{i,j}; B_{i,j})$ distribution, we have for $i, j \in [d]$,

$$\mathbb{E}_{\alpha_{i,j}}[\alpha_{i,j}] = \frac{A_{i,j}}{B_{i,j}}, \tag{A.18}$$

$$\mathbb{E}_{\alpha_{i,j}}[\log(\alpha_{i,j})] = \Upsilon(A_{i,j}) - \log(B_{i,j}), \tag{A.19}$$

where $\Upsilon(\cdot)$ denotes the digamma function. Similarly, we can obtain the required statistics of μ_i.

Because we use an inverse-Gamma$(\Phi_{i,j}, \Psi_{i,j})$ distribution for the variational update of $\beta_{i,j}$,

$$\mathbb{E}_{\beta_{i,j}}\left[\frac{1}{\beta_{i,j}+\Delta_{i,j}(t_{i,n})}\right] = \int_{\mathbb{R}_+} \frac{1}{y+\Delta_{i,j}(t_{i,n})} y^{-\Phi_{i,j}-1}\exp\left(-\Psi_{i,j}/y\right)\frac{dy}{Z},$$
$$\mathbb{E}_{\beta_{i,j}}\Big[\log(\beta_{i,j}+\Delta_{i,j}(t_{i,n}))\Big] = \int_{\mathbb{R}_+} \log(y+\Delta_{i,j}(t_{i,n})) y^{-\Phi_{i,j}-1}\exp\left(-\Psi_{i,j}/y\right)\frac{dy}{Z},$$

where Z denotes the normalization factor of the inverse-Gamma$(\Phi_{i,j}, \Psi_{i,j})$. The above expressions can be approximated as follows

$$\mathbb{E}\left[\frac{1}{\beta_{i,j}+\Delta_{i,j}(t_{i,n})}\right] \approx \frac{1}{\frac{\Psi_{i,j}}{\Phi_{i,j}-1}+\Delta_{i,j}(t_{i,n})}, \tag{A.20}$$

$$\mathbb{E}\left[\log\left(\beta_{i,j}+\Delta_{i,j}(t_{i,n})\right)\right] \approx \log\left(\frac{\Psi_{i,j}}{\Phi_{i,j}-1}+\Delta_{i,j}(t_{i,n})\right). \tag{A.21}$$

A.4.3 Computational Complexity

We report the computational complexity of GB to be $O(|\mathcal{T}|\log d)$, while the authors of the method originally report $O(|\mathcal{T}|(\log|\mathcal{T}|+\log d))$ in [49]. The difference lies in the computation of the inter-event times $\{\Delta_{i,j}(t_{i,n})\}$, where the authors consider the computation of each inter-event time as $O(\log|\mathcal{T}|)$ at each iteration. However, it suffices to precompute these values once and cache them. Therefore, this step is $O(1)$, which reduces the computational complexity of GB to $O(|\mathcal{T}|\log d)$.

B Additional Experimental Results

B.1 Additional Experiments for Chapter 2

We carry out an additional set of experiments to evaluate the effect of zeroing-out small weights using a threshold η. To do so, we first introduce the two performance metrics. The *false positive rate* (FPR) is the fraction of errors in learned edges

$$\text{FPR} = |\{\widehat{w}_{ij} | \widehat{w}_{ij} > 0, w^{\star}_{ij} = 0\}| / |\{\widehat{w}_{ij} | w^{\star}_{ij} = 0\}|,$$

where $|\cdot|$ denotes the cardinality of a set. Similarly, the *false negative rate* (FNR) to be the fraction of errors in learned non-edges

$$\text{FNR} = |\{\widehat{w}_{ij} | \widehat{w}_{ij} = 0, w^{\star}_{ij} > 0\}| \; |\{\widehat{w}_{ij} | w^{\star}_{ij} > 0\}|.$$

Figure B.1 shows the effect of number of samples on F1-score for several choices of threshold η. We see that our proposed algorithm VI-EXP (resp. VI-SG) outperform its MLE counterpart MLE-ADM4 (resp. MLE-SGLP) for all values of η. With increasing η, we see that the F1-score of MLE-based approaches improve. This is due to the FPR decreasing faster than the FNR increases due to the sparsity of the graph. However, note that since we do not know the expected value of true edges w^{*}_{ij} beforehand, it is not clear a-priori what value we should set for the threshold η. Ideally, we choose the threshold η to be as small as possible, which is the regime in which our variational inference algorithm outperforms MLE-based methods the most.

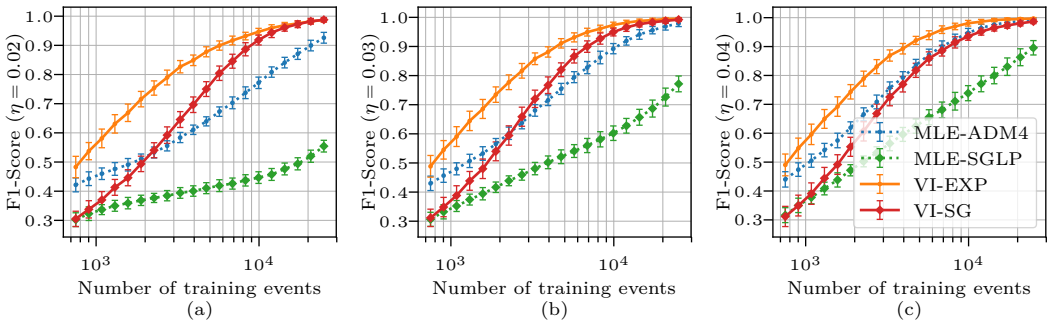


Figure B.1 – Performance measured by F1-Score with respect to the number of training samples for several choices of threshold η. The proposed variational inference approaches are shown in solid lines. The non-parametric methods are highlighted with square markers.

B.2 Additional Experiments for Chapter 3

Similar to Chapter 2, we analyze the effect of zeroing-out small weights using a threshold η. Figure B.2 shows the effect of the variance of the noise on F1-score for several choices of threshold η. We see that the relative performance of both methods remain consistent across all the thresholds considered.

For the sake of completeness and for consistency with the other chapters, we also present the experimental results with respect to the average precision-recall by sweeping over all thresholds $\eta > 0$ in Figure B.3, and we report the precision@k for several values of k in Figure B.4.

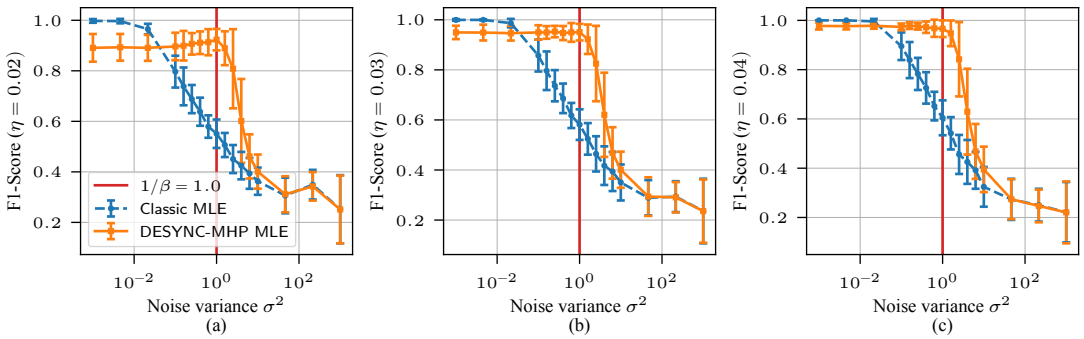


Figure B.2 – Performance measured by F1-Score with respect to the noise scale for several choices of threshold η.

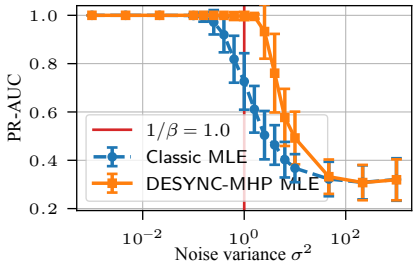


Figure B.3 – Performance measured by PR-AUC, sweeping over all thresholds $\eta > 0$ as in Chapters 4 and 5.

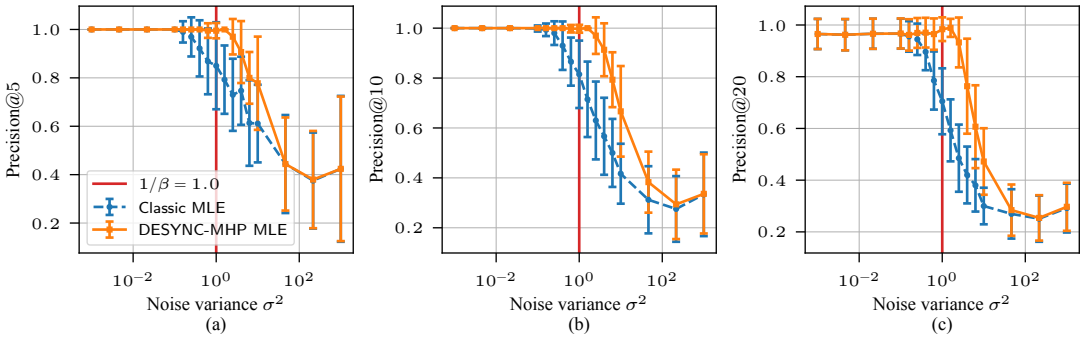


Figure B.4 – Performance measured by precision@k for $k = 5, 10, 20$.

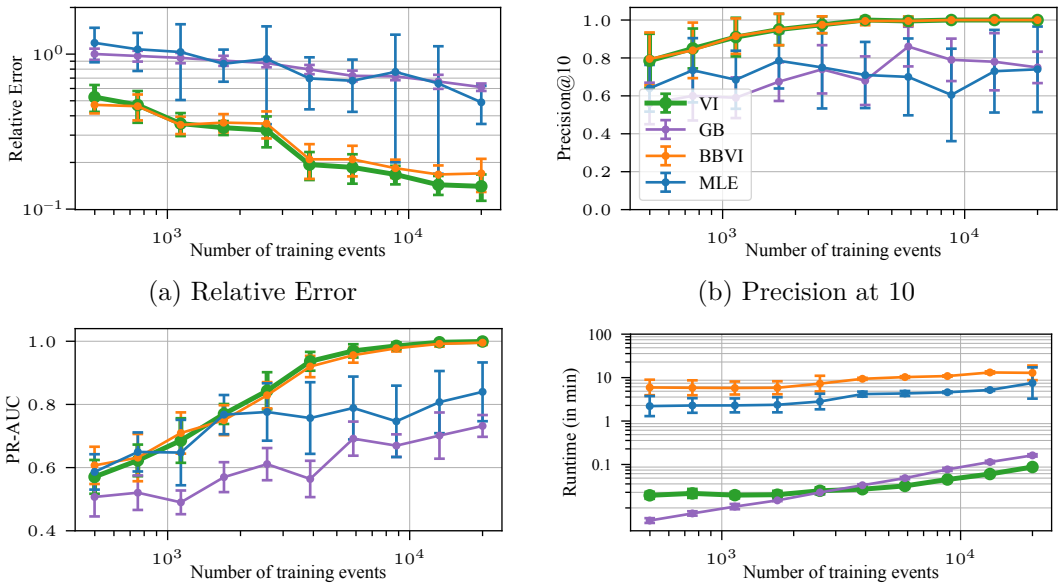
B.3 Additional Experiments for Chapter 5

Analysis of performance w.r.t. number of training events.

To evaluate the number of training samples required to achieve a good performance for each approach, we ran the experiments with the same synthetic simulation setup, fixed the number of dimensions $d = 10$, and varied the number of training events. We present these results in Figure B.5. Although BBVI was originally designed to train on small observations sequences, our VI approach does as well or outperforms BBVI.

Alternative simulation setup.

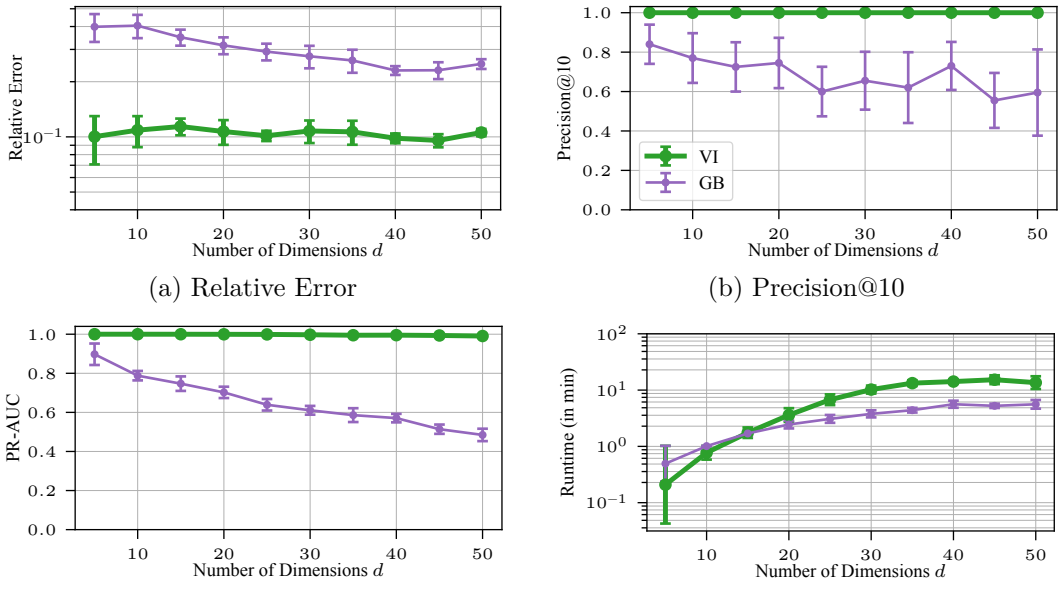
To further investigate the effect of the structural constraint required by GB, *i.e.*, $\sum_i \alpha_{i,j} = 1$ for all $i, j \in [d]$, we ran additional experiments on synthetic data where we normalized the ground-truth $\{\alpha_{i,j}\}$ such that $\sum_i \alpha_{i,j} = 1$. The results are shown in Figure B.6. We see that, even if GB performs better than in Figure 5.2, our VI approach still outperforms GB on all metrics.



(a) Relative Error (b) Precision at 10

(c) PR-AUC (d) Runtime

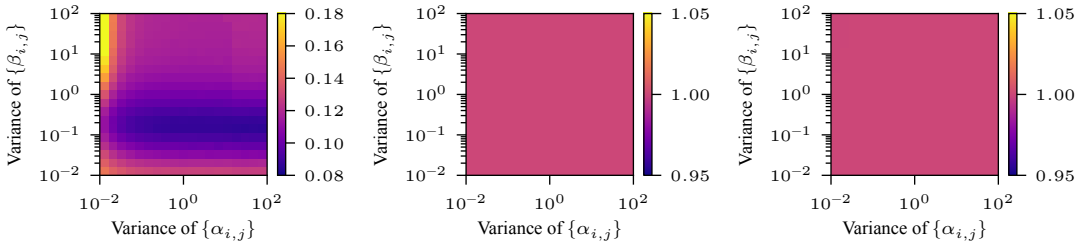
Figure B.5 – Results on synthetic data for varying numbers of training events. Panel (a) (log-scale) relative error, (b) precision@10, (c) PR-AUC, and panel (d) (log-scale) empirical runtime of each approach in minutes.



(a) Relative Error (b) Precision@10

(c) PR-AUC (d) Runtime

Figure B.6 – Results on synthetic data for the alternative synthetic simulation setup where we normalize the $\{\alpha_{i,j}\}$ such that $\sum_i \alpha_{i,j} = 1$. Panel (a) (log-scale) relative error, (b) precision@10, (c) PR-AUC, and panel (d) (log-scale) empirical runtime of each approach in minutes.



(a) Relative Error (b) Precision@10 (c) PR-AUC

Figure B.7 – Analysis of the robustness of VI to the choice of prior. We report the relative error for a wide range of variances for both $\{\alpha_{i,j}\}$, $\{\beta_{i,j}\}$, keeping their mean fixed to the same value used in the experiments.

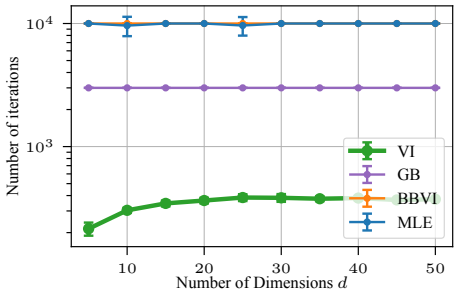


Figure B.8 – Number of iterations performed in the experiments on synthetic data.

Robustness to the choice of prior.

To investigate the sensitivity of VI to choice of the prior, we ran additional experiments on synthetic data. For $d = 10$ dimensions, we fixed the mean as in the experiments of Section 5.5.1, and evaluated the performance for variance of the priors of $\{\alpha_{i,j}\}$ and $\{\beta_{i,j}\}$ ranging between 10^{-2} and 10^2. As seen in Figure B.7, for a large range of priors, VI remains stable. For all values tested, both the PR-AUC and Precision@10 remained at 1.0.

Analysis of the number of iterations.

In Figure 5.2, we discussed the runtime of each algorithm on synthetic data. To make the comparison fair, we also report the number of iterations performed in Figure B.8. As stated in Appendix C.4, we ran VI, BBVI and MLE until convergence or up to maximum 10 000 iterations. As the number of dimensions increases, the number of iterations needed for VI to converge becomes sub-linear. BBVI almost always ran to the cap on the maximum number of iterations because it uses Monte Carlos samples of the posterior at each iteration and hence exhibit a larger variance between iterations. We ran GB for 3000 iterations, which was found to be enough to reach convergence[1].

[1]Note that [49] used 300 iterations without further justification.

C Reproducibility

In this section, we provide extensive details on the experimental setup used in all the experiments of the thesis.

C.1 Experimental setup of Chapter 2

We first describe the implementation details of the algorithm described in Algorithm 2.1. We then provide the details of the experimental setup for both the synthetic and real data experiments. The open-source code used to generate the figures is released publicly on GitHub at `https://github.com/trouleau/var-hawkes`.

C.1.1 Implementation details of Algorithm 2.1

We used $L = 1$ sampled Gaussian noise in line 3 of Algorithm 2.1. We set the momentum term $\zeta = 0.5$ in (2.13). In our early experiments, we observed that the performance of the algorithm is not sensitive to the momentum term ζ for $\zeta \in (0, 1)$. Therefore, we decided to set it to 0.5 in all experiments. We used the Adam optimizer with learning rate $\eta = 0.02$. We also multiply the learning rate by $1 - 10^{-4}$ at each iteration. We initialized $\boldsymbol{\nu}$ by sampling from the normal distribution $\mathcal{N}(0.1, 0.01)$. We initialized $\boldsymbol{\alpha} = 0.1$ for all hyperparameters. We observed that the performance of the algorithm is not sensitive to the initialization. We initialized $\boldsymbol{\sigma}$ by sampling from the normal distribution $\mathcal{N}(0.2, 0.01)$ then clipping them to be in $[0.01, 2]$. This initialization ensures that the initial variance of the algorithm is neither too small nor too big.

C.1.2 Synthetic Data

To create the synthetic data, we generated random Erdős–Rényi graphs with $d = 50$ nodes and with edge probability $p = \log(d)/d$, leading to graphs with 195 edges on average.

Then, the sequences of observations were generated from a multivariate Hawkes process with exponential excitation kernels defined in (2.1). The baseline $\{\mu_i\}$ were sampled uniformly at random in $[0, 0.02]$, and the edge weights $\{w^*_{ij}\}$ were sampled uniformly at random in $[0.1, 0.2]$. To generate the results of Figure 2.1, we varied the length of observations between $N = 700$ and $N = 25000$. The results were averaged over 30 graphs with 10 simulations each.

We used the Python library `tick` to run the MLE-based baseline approaches [12]. To tune the hyperparameters of the MLE-based approaches, we first manually searched for an initial range of parameters where the algorithm performed well. Then, we fined-tuned the hyperparameters using grid-search to find the ones giving the best results for the Precision@20 and F1-score metrics. For MLE-SGLP, we used the grid range $1/\alpha \in [0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0]$ and `lasso_grouplasso_ratio` $\in [0.25, 0.5, 0.75]$. We used the default values for the optimizer, which we checked and are sure of its convergence. We finally chose $1/\alpha = 0.1$ and `lasso_grouplasso_ratio` $= 0.75$. For MLE-ADM4, we also used the grid range $1/\alpha \in [0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0]$ and lasso_nuclear_norm $\in [0.25, 0.5, 0.75]$. Making overall 21 different configurations. We finally chose $1/\alpha = 0.05$ and lasso_nuclear_norm $= 0.5$ that gave the best results for Precision@20 and F1-score.

C.1.3 Real Data

For our approach VI-EXP and its parametric counterpart VI-SG, the exponential decay parameter must be tuned for each dataset. As expected, both algorithms performed best with the same decay. For our approach VI-SG and its MLE counterpart MLE-SGLP, there are two parameters to tune, M and cutoff time T_c. The center of the m-th Gaussian kernel, with $m \in [M]$, is defined as $t_m = T_c \cdot (m-1)/M$ and its scale is defined as $b = T_c/(\pi \cdot M)$ in (2.2). After manually finding an initial range of M and T where algorithms performed well, we then fine-tuned them using grid-search.

Epidemic dataset. This dataset is publicly available in the supplementary material of [51]. We performed the following hyperparameter search. For our VI-SG algorithm, we did a grid-search with $M \in \{30, 35, 40, 45, 50, 55\}$ and $T_c \in \{025 \cdot M, 0.5 \cdot M, 0.75 \cdot M\}$. We did not see a notable difference between the performance of different grids, as long as M and T are large enough. We chose $M = 55$ and $T = 27.5$. For the baseline MLE-SGLP, we did a grid-search with $M \in \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21\}$, $T_c \in \{0.25M, 0.5M, 1M, 2M, 5M, 10M, 20M, 40M\}$ and $1/\alpha \in \{1, 10, 50, 100\}$, that makes overall 352 experiments. We chose $M = 19$, $T_c = 9.5$ and $1/\alpha = 10$. For our algorithm VI-EXP, we tried decay $\in [0.1, 0.5, 1, 2, 5, 10, 20, 40]$ and we chose decay $= 0.1$. For the baseline MLE-ADM4, we did a grid-search with decay $\in \{0.1, 0.5, 1, 2, 5, 10, 20, 40\}$ and $1/\alpha = \{0.01, 0.1, 1, 2, 5, 10, 50, 100, 200, 400, 800\}$. We chose decay $= 0.1$ and $1/\alpha = 50$.

Stock market dataset. This dataset was generously provided to us by the authors of [44]. In the stock market dataset, our algorithm VI-SG also performed better with a larger M. As for large M the experiments are slow we decided to set $M = 50$ and did grid-search for T_c with $T_c \in [0.15 \cdot M, 0.25 \cdot M, 0.5 \cdot M]$. For the baseline MLE-SGLP, we did a grid-search with $M \in \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21\}$, $T_c \in \{0.25 \cdot M, 0.5 \cdot M, 0.75 \cdot M, 1 \cdot M, 2 \cdot M, 5 \cdot M\}$ and $1/\alpha \in \{0.01, 0.1, 0.5, 1, 10, 50, 100\}$. The best values found were $M = 17$, $T_c = 8.5$ and $C = 0.1$. For our algorithm VI-EXP, we tried decay $\in \{0.1, 0.5, 1, 2, 5, 10, 20, 40\}$ and we chose decay = 0,1. For the baseline MLE-ADM4, we did a grid search with decay $\in \{0.1, 0.5, 1, 2, 5, 10, 20, 40\}$ and $1/\alpha = \{0.01, 0.1, 1, 2, 5, 10, 50, 100, 200, 400, 800\}$. We chose decay = 0,1 and $1/\alpha = 1$.

Enron email dataset. The Enron email dataset is available at: `https://www.cs.cmu.edu/~enron/`. Preprocessing details are made available online at `https://github.com/trouleau/var-hawkes`. Because it is a larger dataset and experiments are more computationally intensive, we chose smaller ranges for hyperparameter tuning. For our algorithm VI-SG we did a grid-search with $M = 10$ and $T_c \in [5, 7.5, 10, 15]$. The best value is $T_c = 5$. For the baseline MLE-SGLP, we did a grid-search with $M \in \{1, 2, 3, 4, 5\}$, $T_c \in \{0.1, 0.25, 0.5, 0.75, 1, 1.25\}$ and $1/\alpha \in \{10, 20, 50, 100, 500\}$. The best value is $M = 1$, $T_c = 2.5$ and $1/\alpha = 50$. For our algorithm VI-EXP, we tried decay $\in \{5, 10, 20, 40\}$ and we chose decay = 20. For the baseline MLE-ADM4, we did a grid-search with decay $\in \{0.1, 0.5, 1, 2, 5, 10, 20, 40\}$ and $1/\alpha = \{0.01, 0.1, 1, 2, 5, 10, 50, 100, 200, 400, 800\}$. We chose decay = 20 and $1/\alpha = 0.1$.

C.2 Experimental setup of Chapter 3

In this section, we provide details of the experimental setup used to produce the figures reported in Chapter 3. The open-source code used to generate the figures are released publicly on GitHub at `https://github.com/trouleau/desync-mhp`.

C.2.1 Synthetic Data

We set the exponential decay to $\beta = 1.0$. For smoothing, we ran experiments for a wide range of hyperparameters β' and γ in $[10^0, 10^3]$ such that $\beta' < \gamma$. We found that the performance of the algorithm is not sensitive to the choice of hyperparameters as long as $\beta \ll \beta' \ll \gamma$ to satisfy the assumptions of the approximation in (3.5). In all experiments, we used $\beta' = 50$ and $\gamma = 500$. We used the same L1 penalty with the same weight $1/\alpha_{\boldsymbol{\theta}} = 5000$ for both methods. For DESYNC-MHP, we also use a L2 penalty for the noise parameters with weight $1/\alpha_{\boldsymbol{z}} = 2000$.

For each experiment, we chose small positive background intensities $\mu_i = 0.05 \forall i \in [d]$ and generated a random excitation matrices with entries $\{w_{i,j}\} \in \{0, 1\}$ by sampling edges independently with probability $2/d$. The average in-degree and out-degree of each node was hence close to two. We then rescaled the entries to obtain a spectral radius of 0.95 to ensure that the simulated processes are stable. Experiments were not found to be sensitive to this choice of value. We generated $C = 5$ realizations of $50,000$ samples from the Hawkes process using Ogata's thinning algorithm with the Python library *tick* from Bacry et al. [12]. We repeated each experiment 10 times over 10 different matrices for each set of parameters.

C.2.2 Real Data

For computational reasons, hyperparameter search was performed on the classic ML estimator. Both methods assume an exponential excitation function with a fixed exponential decay that needs to be tuned. We tuned the exponential decay β using grid-search to maximize the log-likelihood of the classic ML estimator. In all experiments, we used $\beta = 0.0047$. We used the same L1 penalty with the same weight $1/\alpha_{\boldsymbol{\theta}}$ for both methods. Similar to the exponential decay, we tuned the penalty weight using grid-search with $1/\alpha_{\boldsymbol{\theta}} \in \{0.1, 0.5, \ldots, 10^5, 5 \times 10^5\}$ and set it to $1/\alpha_{\boldsymbol{\theta}} = 5000$. For DESYNC-MHP, we also use a L2 penalty for the noise parameters. We did a grid-search with $1/\alpha_{\boldsymbol{z}} \in \{1000, 2000, \ldots, 5000\}$ and chose $1/\alpha_{\boldsymbol{z}} = 2000$. Similar to the synthetic experiments, the performance of the algorithm was not found to be sensitive to the choice of the hyperparameters β' and γ for the smooth approximation of the excitation function. We used $\beta' = 0.16$ and $\gamma = 10\beta' = 1.6$ using grid-search on β'.

C.3 Experimental setup of Chapter 4

In this section, we provide details of the experimental setup used to produce the figures reported in Chapter 4. The open-source code used to generate the figures is released publicly on GitHub.

C.3.1 Synthetic data

The experimental setup for the experiments on synthetic data is as follows. We simulated 20 datasets, each comprised of 5 realizations of 10^5 events. We then randomly translated each dataset with distributions $F_i \sim \mathcal{N}(0, \sigma^2),\ 1 \leq i \leq d$, for 20 noise powers σ^2 ranging from 0.1 to 25, sampled in log-space.

The hyperparameters used to produce the figures in Section 4.7 can be found in notebook and scripts. Details are as follows.

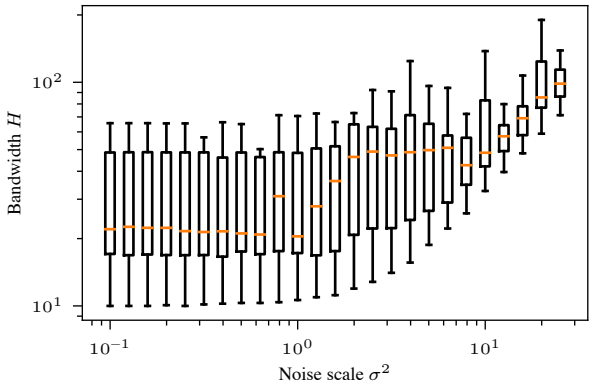


Figure C.1 – Tuning of the bandwidth H of the cumulants estimator used in the experiments of NPHC. We observe that the for noise power in range $\sigma^2 < 1/\beta = 1$, the best bandwidth H remains stable. For $\sigma^2 < 1/\beta = 1$ the best bandwidth H was increases linearly with σ^2.

- ADM4. The exponential decay was set to its ground-truth value, $\beta = 1$.
- Desync-MLE. Similar to ADM4, the exponential decay was set to its ground-truth value, $\beta = 1$.
- NPHC. The bandwidth H of the estimator of the cumulants was set using binary search to minimize the $L_{2,2}$ distance to the ground-truth cumulants. The resulting bandwidth used to run the algorithm are discussed in Figure C.1. In short, we observed that the for noise powers $\sigma^2 < 1/\beta = 1$, the best bandwidth H remained stable. For $\sigma^2 < 1/\beta = 1$ the best bandwidth H increased linearly with σ^2.
- WH. The maximum support of the excitation matrix was set to 10.0 to roughly match the same scale as the ground-truth excitation functions. The number of quadrature points was set to 20. This value, which has a quadratic cost in the computational complexity of the algorithm, was found to be large enough to provide prefect PR-AUC and Precision@k on noiseless observations.

C.3.2 Real data

The dataset used in the experiments is that of Bund Futures traded at Eurex over 20 days in April 2014 [1]. This dataset has already been modeled using Hawkes processes in [9]. Each day is considered an independent realization of the process. The timestamps are recorded at the microsecond timestamp resolution. As explained on the download website, the data was preprocessed as follows. Market opens at 8AM which corresponds to a timestamp of 28 800. This timestamp has been subtracted to all timestamps to

[1] Dataset available at: `https://github.com/X-DataInitiative/tick-datasets/`.

have a realization that starts at time 0. As markets closes at 10PM, the end time of the realizations is 50 400. No additional preprocessing was performed on the dataset.

The first 5 days were used to tune the hyperparameters of the learning algorithms, and the remaining 15 days were used to measure the performance reported in Figure 4.6.

The hyperparameters used to produce the figures in Section 4.7 can be found in notebook and scripts. Details are as follows.

- ADM4. To set the exponential decay β of the excitation functions, we ran a grid search for values between 10^2 and 10^4, and we found that $\beta = 1291.0$ maximized the log-likelihood. We used grid search between 1 and 10^5 to tune the regularization weight and found that 10^3 maximized the log-likelihood.

- NPHC. Following the observation made in the experiments on synthetic data, the bandwidth H of the estimator of the cumulants, was set to $H = 1/\beta + \sigma^2$. This value provided stable results.

- WH. The hyperparameters of this algorithm were set as in [9].

We were unable to reproduce the results of [9] on the noiseless dataset (*i.e.*, without added random translation) using Desync-MLE. Since the dataset consists of timestamps in a high-frequency trading application, it is not expected to hold any synchronization noise. However, as shown in Figure C.2, Desync-MLE converged to a non-zero synchronization noise with a diagonal excitation matrix.

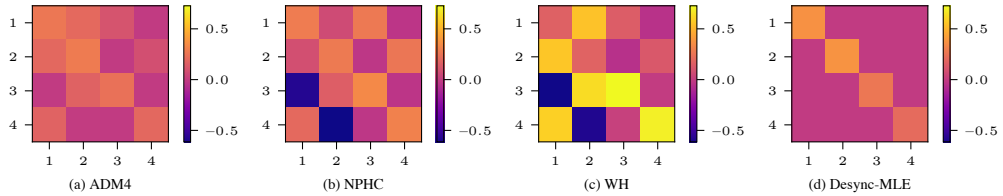


Figure C.2 – Excitation matrices $\widehat{\mathbf{\Phi}}_0$ learned by the different learning methods on the noiseless dataset of Bund Futures traded at Eurex.

C.4 Experimental setup of Chapter 5

In this section, we provide details of the experimental setup used to produce the figures reported in Chapter 3. The open-source code used to generate the figures is released publicly on GitHub at `https://github.com/trouleau/var-wold`.

C.4.1 Simulation setup for synthetic data

We generated Erdős–Rényi random graphs with d nodes. We sampled background rates $\{\mu_i^*\}$ from Uniform$[0, 0.05]$, edge weights $\{\alpha_{i,j}^*\}$ from Uniform$[0.1, 0.2]$ for all edges, and parameters $\{\beta_{i,j}^*\}$ from Uniform$[1, 2]$, all independently. Each algorithm was then run as follows.

VI. We ran the algorithm for a maximum of 10 000 iterations or until convergence. We defined convergence when the maximum absolute difference of any parameter between two consecutive iterations is less than 10^{-4}. We used priors $p(\mu_i) = \text{Gamma}(0.1, 1)$, $p(\alpha_{i,j}) = \text{Gamma}(0.1, 1)$ and $p(\beta_{i,j}) = \text{InverseGamma}(100, 100)$.

GB. We used the implementation released in [49]. We used 3000 iterations in all experiments. As advised by the authors, we used the same Dirichlet prior with uniform parameters $1/d$, and set the parameters $\{\beta_{i,j}\}$ to the data-driven heuristic $\beta_{i,j} = \text{median}(\{t_{i,n} - t_{i,n-1} | t_{i,n} \in \mathcal{T}_i\})/\exp(1)$.

BBVI. We adapted the method introduced in Chapter 2 with the intensity of a Wold process. Analogous to VI, we ran the method for a maximum of 10 000 iterations or until convergence. As in Chapter 2, we used Log-Normal posterior distributions, Laplacian priors $\{\alpha_{i,j}\}$, and Gaussian priors for $\{\mu_i\}$ and $\{\beta_{i,j}\}$ with the same parameters.

MLE. Analogous to VI, we ran the method for a maximum of 10 000 iterations or until convergence.

All experiments were run on a single-core, on the same machine with a processor *Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz* and 256 GB of RAM.

C.4.2 Experiments on Real Datasets

Email-EU-core dataset

As explained in Section 5.5, the Email-EU-core dataset is composed of emails between researchers from a European research institution. Each email in the dataset is a tuple (sender, receiver, timestamp). To build each process from the dataset, we used the same preprocessing steps as Figueiredo et al. [49]. More precisely, we excluded users with no sent email and defined the set of processes as the top-100 users with the most received emails. We then aggregated the timestamps by receivers. The entries in the ground-truth influence matrix are defined by counting the number of emails sent from each sender to each receiver (a weight zero indicates the absence of an edge). The preprocessing code is made available publicly.

For the hyperparameters, we ran a sweep over the Dirichlet prior of GB over $[0.01, 0.1, 1.0, 10.0, 100.0]$ and reported the best results obtained with 10.0. For VI, we ran a sweep over the of parameters of the priors over $[0.01, 0.1, 1.0, 10.0, 100.0]$ and used $p(\mu_i) = \text{Gamma}(1.0, 1.0)$, $p(\alpha_{i,j}) = \text{Gamma}(1.0, 1.0)$ and $p(\beta_{i,j}) = \text{Inverse}-\text{Gamma}(100.0, 100.0)$.

MemeTracker dataset.

The MemeTracker dataset is composed of online blog posts. We used the top-100 blogs with the highest number of published posts and built the processes by aggregating the sequences of published timestamps, resulting in 15 168 774 events in 100 dimensions. The preprocessing code is available publicly[2]. We ran a sweep over the Dirichlet prior of GB over $[0.01, 0.1, 1.0, 10.0]$, and did not observe a significant difference between the different values and reported the results obtained for 0.01. For VI, we used priors $p(\mu_i) = \text{Gamma}(0.1, 1)$, $p(\alpha_{i,j}) = \text{Gamma}(0.1, 1)$ and $p(\beta_{i,j}) = \text{Inverse}-\text{Gamma}(10^4, 10^4)$.

[2]The code is made publicly available at `https://github.com/achab/nphc/tree/master/nphc/datasets/memetracker` by the authors of [2].

Bibliography

[1] M. Achab. *Learning from Sequences with Point Processes*. Theses, Université Paris Saclay (COmUE), Oct. 2017. URL `https://pastel.archives-ouvertes.fr/tel-01775239`. [Cited on page 67]

[2] M. Achab, E. Bacry, S. Gaïffas, I. Mastromatteo, and J.-F. Muzy. Uncovering causality from multivariate Hawkes integrated cumulants. *The Journal of Machine Learning Research*, 18(1):6998–7025, 2017. [Cited on pages 19, 26, 34, 58, 61, 64, 66, 74, 83, 84, 98, 99, and 120]

[3] R. A. d. S. Alves, R. M. Assuncao, and P. O. S. Vaz de Melo. Burstiness scale: A parsimonious model for characterizing random series of events. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1405–1414, 2016. [Cited on page 75]

[4] I. Apostolopoulou, S. Linderman, K. Miller, and A. Dubrawski. Mutually regressive point processes. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/564645fbd0332f066cbd9d083ddd077c-Paper.pdf`. [Cited on pages 20 and 21]

[5] P. M. Asaro. Ai ethics in predictive policing: From models of threat to an ethics of care. *IEEE Technology and Society Magazine*, 38(2):40–53, 2019. doi: 10.1109/MTS.2019.2915154. [Cited on page 21]

[6] Y. Aït-Sahalia, J. Cacho-Diaz, and R. J. Laeven. Modeling financial contagion using mutually exciting jump processes. *Journal of Financial Economics*, 117(3):585–606, 2015. ISSN 0304-405X. doi: https://doi.org/10.1016/j.jfineco.2015.03.002. URL `https://www.sciencedirect.com/science/article/pii/S0304405X15000264`. [Cited on page 20]

[7] E. Bacry and J.-F. Muzy. First- and second-order statistics characterization of Hawkes processes and non-parametric estimation. *IEEE Transactions on Information Theory*, 62(4):2184–2202, 2016. [Cited on pages 18, 58, 65, 66, and 74]

[8] E. Bacry, K. Dayri, and J.-F. Muzy. Non-parametric kernel estimation for symmetric Hawkes processes. application to high frequency financial data. *The European Physical Journal B-Condensed Matter and Complex Systems*, 85(5):1–12, 2012. [Cited on page 58]

[9] E. Bacry, T. Jaisson, and J.-F. Muzy. Estimation of slowly decreasing Hawkes kernels: Application to high frequency order book modelling. *arXiv preprint arXiv:1412.7096*, 2014. [Cited on pages 117 and 118]

[10] E. Bacry, S. Gaïffas, and J.-F. Muzy. A generalization error bound for sparse and low-rank multivariate Hawkes processes. *arXiv preprint arXiv:1501.00725*, 2015. [Cited on pages 18 and 28]

[11] E. Bacry, I. Mastromatteo, and J.-F. Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015. [Cited on pages 2, 17, 20, and 43]

[12] E. Bacry, M. Bompaire, S. Gaïffas, and S. Poulsen. tick: a Python library for statistical learning, with a particular emphasis on time-dependent modeling. *ArXiv e-prints*, July 2017. [Cited on pages 50, 114, and 116]

[13] R. Bamler and S. Mandt. Dynamic word embeddings. In *Proceedings of the 34th international conference on Machine learning*, 2017. [Cited on page 27]

[14] R. Bamler, F. Salehi, and S. Mandt. Augmenting and tuning knowledge graph embeddings. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2019. [Cited on pages 27 and 32]

[15] A. Bar-Hen, J. Chadœuf, H. Dessard, and P. Monestiez. Estimating second order characteristics of point processes with known independent noise. *Statistics and Computing*, 23(3):297–309, May 2013. ISSN 1573-1375. doi: 10.1007/s11222-011-9311-7. URL `https://doi.org/10.1007/s11222-011-9311-7`. [Cited on pages 41, 42, and 58]

[16] O. Barkan. Bayesian neural word embedding. In *AAAI*, pages 3135–3143, 2017. [Cited on page 27]

[17] J. Bernardo, M. Bayarri, J. Berger, A. Dawid, D. Heckerman, A. Smith, M. West, et al. The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian statistics*, 7:453–464, 2003. [Cited on page 27]

[18] A. L. Bertozzi, E. Franco, G. Mohler, M. B. Short, and D. Sledge. The challenges of modeling and forecasting the spread of covid-19. *Proceedings of the National Academy of Sciences*, 117(29):16732–16738, 2020. ISSN 0027-8424. doi: 10.1073/

pnas.2006520117. URL https://www.pnas.org/content/117/29/16732. [Cited on pages 19 and 88]

[19] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. [Cited on pages 27 and 30]

[20] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. [Cited on page 100]

[21] C. Blundell, J. Beck, and K. A. Heller. Modelling reciprocating relationships with hawkes processes. In *Advances in Neural Information Processing Systems*, pages 2600–2608, 2012. [Cited on page 20]

[22] C. G. Bowsher. Modelling security market events in continuous time: Intensity based, multivariate point process models. *Journal of Econometrics*, 141(2):876–912, 2007. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2006.11.007. URL https://www.sciencedirect.com/science/article/pii/S030440760600251X. [Cited on page 20]

[23] P. J. Brantingham, M. Valasik, and G. O. Mohler. Does predictive policing lead to biased arrests? results from a randomized controlled trial. *Statistics and Public Policy*, 5(1):1–6, Jan. 2018. doi: 10.1080/2330443x.2018.1438940. URL https://doi.org/10.1080/2330443x.2018.1438940. [Cited on page 21]

[24] P. Brémaud. *Fourier Analysis and Stochastic Processes*. Springer International Publishing, 2014. doi: 10.1007/978-3-319-09590-5. URL https://doi.org/10.1007/978-3-319-09590-5. [Cited on page 17]

[25] P. Brémaud. *Point Process Calculus in Time and Space*. Springer International Publishing, 2020. doi: 10.1007/978-3-030-62753-9. URL https://doi.org/10.1007/978-3-030-62753-9. [Cited on page 17]

[26] R. T. Q. Chen, B. Amos, and M. Nickel. Neural spatio-temporal point processes, 2021. [Cited on page 88]

[27] S. Chen, A. Shojaie, E. Shea-Brown, and D. Witten. The multivariate hawkes process in high dimensions: Beyond mutual excitation, 2019. [Cited on pages 20 and 21]

[28] W.-H. Chiang, X. Liu, and G. Mohler. Hawkes process modeling of covid-19 with mobility leading indicators and spatial covariates. *medRxiv*, 2020. doi: 10.1101/2020.06.06.20124149. URL https://doi.org/10.1101/2020.06.06.20124149. [Cited on pages 19 and 88]

[29] E. Choi, N. Du, R. Chen, L. Song, and J. Sun. Constructing disease network and temporal progression model via context-sensitive Hawkes process. In *Proceedings*

of the 2015 IEEE International Conference on Data Mining (ICDM), ICDM '15, pages 721–726, Washington, DC, USA, 2015. IEEE Computer Society. ISBN 978-1-4673-9504-5. doi: 10.1109/ICDM.2015.144. URL http://dx.doi.org/10.1109/ICDM.2015.144. [Cited on page 19]

[30] D. R. Cox. Some statistical methods connected with series of events. *Journal of the Royal Statistical Society: Series B (Methodological)*, 17(2):129–157, 1955. [Cited on pages 9 and 74]

[31] C. Cremer, Q. Morris, and D. Duvenaud. Reinterpreting importance-weighted autoencoders. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Syw2ZgrFx. [Cited on page 30]

[32] L. Cucala. Intensity estimation for spatial point processes observed with noise. *Scandinavian Journal of Statistics*, 35:322–334, 06 2008. doi: 10.1111/j.1467-9469.2007.00583.x. [Cited on pages 41, 42, and 58]

[33] D. Daley. Stationary point processes by markov-dependent intervals and infinite intensity. *Journal of Applied Probability*, 19(A):313–320, 1982. [Cited on pages 9 and 74]

[34] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes. Vol. I.* Probability and its Applications (New York). Springer-Verlag, New York, second edition, 2003. ISBN 0-387-95541-0. Elementary theory and methods. [Cited on pages 3, 6, 9, 13, 17, 28, 60, 73, 74, 76, 77, and 93]

[35] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes. Vol. II.* Probability and its Applications (New York). Springer, New York, second edition, 2008. ISBN 978-0-387-21337-8. URL http://www.springerlink.com/content/978-0-387-21337-8. General theory and structure. [Cited on page 3]

[36] A. De, I. Valera, N. Ganguly, S. Bhattacharya, and M. G. Rodriguez. Learning and forecasting opinion dynamics in social networks. In *NIPS '16: Advances in Neural Information Processing Systems*, 2016. [Cited on page 20]

[37] I. Deutsch and G. J. Ross. Abc learning of hawkes processes with missing or noisy event times, 2021. [Cited on page 59]

[38] V. Didelez. Graphical models for marked point processes based on local independence. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):245–264, 2008. doi: https://doi.org/10.1111/j.1467-9868.2007.00634.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00634.x. [Cited on page 15]

[39] J. L. Doob. *Stochastic processes*, volume 101. New York Wiley, 1953. [Cited on page 93]

[40] N. Du, L. Song, M. Gomez Rodriguez, and H. Zha. Scalable influence estimation in continuous-time diffusion networks. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/8fb21ee7a2207526da55a679f0332de2-Paper.pdf. [Cited on page 20]

[41] N. Du, M. Farajtabar, A. Ahmed, A. J. Smola, and L. Song. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, page 219–228, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336642. doi: 10.1145/2783258.2783411. URL https://doi.org/10.1145/2783258.2783411. [Cited on page 19]

[42] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1555–1564, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939875. URL https://doi.org/10.1145/2939672.2939875. [Cited on page 21]

[43] M. Eichler, R. Dahlhaus, and J. Dueck. Graphical modeling for multivariate hawkes processes with nonparametric link functions. *Journal of Time Series Analysis*, 38(2):225–242, 2017. doi: https://doi.org/10.1111/jtsa.12213. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/jtsa.12213. [Cited on pages 14, 15, and 76]

[44] J. Etesami, N. Kiyavash, K. Zhang, and K. Singhal. Learning network of multivariate Hawkes processes: a time series approach. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'16, pages 162–171, Arlington, Virginia, United States, 2016. AUAI Press. ISBN 978-0-9966431-1-5. URL http://dl.acm.org/citation.cfm?id=3020948.3020966. [Cited on pages 15, 18, 37, 58, 76, and 115]

[45] J. Etesami*, W. Trouleau*, N. Kiyavash, M. Grossglauser, and P. Thiran. A variational inference approach to learning multivariate Wold processes. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 130 of *Proceedings of Machine Learning Research*. PMLR, 2021. [Cited on page 73]

[46] M. Farajtabar, N. Du, M. Gomez Rodriguez, I. Valera, H. Zha, and L. Song. Shaping social activity by incentivizing users. In Z. Ghahramani,

M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/996009f2374006606f4c0b0fda878af1-Paper.pdf. [Cited on page 20]

[47] M. Farajtabar, N. Du, M. Gomez Rodriguez, I. Valera, H. Zha, and L. Song. Shaping social activity by incentivizing users. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2474–2482. Curran Associates, Inc., 2014. [Cited on pages 18, 43, and 46]

[48] M. Farajtabar, M. Gomez-Rodriguez, Y. Wang, S. Li, H. Zha, and L. Song. Coevolve: A joint point process model for information diffusion and network co-evolution. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 473–477, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee. ISBN 9781450356404. doi: 10.1145/3184558.3186236. URL https://doi.org/10.1145/3184558.3186236. [Cited on pages 2 and 20]

[49] F. Figueiredo, G. Borges, P. O. S. V. de Melo, and R. Assunção. Fast estimation of causal interactions using Wold processes. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 2975–2986, USA, 2018. Curran Associates Inc. URL http://dl.acm.org/citation.cfm?id=3327144.3327220. [Cited on pages 35, 69, 74, 75, 77, 81, 82, 83, 84, 105, 111, and 119]

[50] E. W. Fox, M. B. Short, F. P. Schoenberg, K. D. Coronges, and A. L. Bertozzi. Modeling e-mail networks and inferring leadership using self-exciting point processes. *Journal of the American Statistical Association*, 111(514):564–584, 2016. doi: 10.1080/01621459.2015.1135802. URL https://doi.org/10.1080/01621459.2015.1135802. [Cited on page 20]

[51] T. Garske, A. Cori, A. Ariyarajah, I. M. Blake, I. Dorigatti, T. Eckmanns, C. Fraser, W. Hinsley, T. Jombart, H. L. Mills, G. Nedjati-Gilani, E. Newton, P. Nouvellet, D. Perkins, S. Riley, D. Schumacher, A. Shah, M. D. V. Kerkhove, C. Dye, N. M. Ferguson, and C. A. Donnelly. Heterogeneities in the case fatality ratio in the West African ebola outbreak 2013-2016. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1721):20160308, 2017. doi: 10.1098/rstb.2016.0308. URL https://royalsocietypublishing.org/doi/abs/10.1098/rstb.2016.0308. [Cited on pages 37 and 114]

[52] F. Gerhard, M. Deger, and W. Truccolo. On the stability and dynamics of stochastic spiking neuron models: Nonlinear hawkes process and point process glms. *PLOS Computational Biology*, 13(2):1–31, 02 2017. doi: 10.1371/journal.pcbi.1005390. URL https://doi.org/10.1371/journal.pcbi.1005390. [Cited on pages 20 and 21]

[53] C. W. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, pages 424–438, 1969. [Cited on page 14]

[54] B. Green, T. Horel, and A. V. Papachristos. Modeling contagion through social networks to explain and predict gunshot violence in chicago, 2006 to 2014. *JAMA Internal Medicine*, 177(3):326, Mar. 2017. doi: 10.1001/jamainternmed.2016.8245. URL `https://doi.org/10.1001/jamainternmed.2016.8245`. [Cited on page 21]

[55] G. Gusto and S. Schbath. FADO: A statistical method to detect favored or avoided distances between occurrences of motifs using the hawkes' model. *Statistical Applications in Genetics and Molecular Biology*, 4(1), Jan. 2005. doi: 10.2202/1544-6115.1119. URL `https://doi.org/10.2202/1544-6115.1119`. [Cited on page 19]

[56] P. Guttorp and T. L. Thorarinsdottir. What happened to discrete chaos, the quenouille process, and the sharp markov property? some history of stochastic point processes. *International Statistical Review*, 80(2):253–268, 2012. [Cited on pages 8 and 74]

[57] N. R. Hansen, P. Reynaud-Bouret, and V. Rivoirard. Lasso and probabilistic inequalities for multivariate point processes. *Bernoulli*, 21(1):83–143, 2015. doi: 10.3150/13-BEJ562. URL `https://hal.archives-ouvertes.fr/hal-00722668`. 61 pages. [Cited on pages 26 and 27]

[58] A. G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971. [Cited on pages 2, 9, 19, 58, 65, and 74]

[59] J. Hoffmann and C. Caramanis. Learning graphs from noisy epidemic cascades. *Proc. ACM Meas. Anal. Comput. Syst.*, 3(2), June 2019. doi: 10.1145/3341617.3326155. URL `https://doi.org/10.1145/3341617.3326155`. [Cited on page 59]

[60] V. Isham. A markov construction for a multidimensional point process. *Journal of Applied Probability*, 14(3):507–515, 1977. [Cited on page 74]

[61] G. Ji, R. Bamler, E. B. Sudderth, and S. Mandt. Bayesian paragraph vectors. *Symposium on Advances in Approximate Bayesian Inference*, 2017. [Cited on page 27]

[62] S. Jovanović, J. Hertz, and S. Rotter. Cumulants of hawkes point processes. *Physical Review E*, 91(4):042802, 2015. [Cited on pages 13, 16, 61, and 95]

[63] J. D. Kelly, J. Park, R. J. Harrigan, N. A. Hoff, S. D. Lee, R. Wannier, B. Selo, M. Mossoko, B. Njoloko, E. Okitolonda-Wemakoy, P. Mbala-Kingebeni, G. W. Rutherford, T. B. Smith, S. Ahuka-Mundeke, J. J. Muyembe-Tamfum, A. W. Rimoin, and F. P. Schoenberg. Real-time predictions of the 2018–2019 ebola virus disease outbreak in the democratic republic of the congo using hawkes

point process models. *Epidemics*, 28:100354, 2019. ISSN 1755-4365. doi: https://doi.org/10.1016/j.epidem.2019.100354. URL https://www.sciencedirect.com/science/article/pii/S1755436519300258. [Cited on page 19]

[64] H. Kim. *Spatio-temporal point process models for the spread of avian influenza virus (H5N1)*. PhD thesis, UC Berkeley, 2011. [Cited on page 19]

[65] M. Kim, D. Paini, and R. Jurdak. Modeling stochastic processes in disease spread across a heterogeneous social system. *Proceedings of the National Academy of Sciences*, 116(2):401–406, Dec. 2018. doi: 10.1073/pnas.1801429116. URL https://doi.org/10.1073/pnas.1801429116. [Cited on pages 2 and 19]

[66] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014. [Cited on pages 31 and 32]

[67] M. Kirchner. Hawkes and INAR(∞) processes. *Stochastic Processes and their Applications*, 126(8):2494–2525, Aug. 2016. doi: 10.1016/j.spa.2016.02.008. URL https://doi.org/10.1016/j.spa.2016.02.008. [Cited on page 88]

[68] M. Kirchner. An estimation procedure for the hawkes process. *Quantitative Finance*, 17(4):571–595, Sept. 2016. doi: 10.1080/14697688.2016.1211312. URL https://doi.org/10.1080/14697688.2016.1211312. [Cited on page 88]

[69] E. L. Lai, D. Moyer, B. Yuan, E. Fox, B. Hunter, A. L. Bertozzi, and P. J. Brantingham. Topic time series analysis of microblogs. *IMA Journal of Applied Mathematics*, 81(3):409–431, 07 2016. ISSN 0272-4960. doi: 10.1093/imamat/hxw025. URL https://doi.org/10.1093/imamat/hxw025. [Cited on page 19]

[70] R. C. Lambert, C. Tuleau-Malot, T. Bessaih, V. Rivoirard, Y. Bouret, N. Leresche, and P. Reynaud-Bouret. Reconstructing the functional connectivity of multiple spike trains using hawkes models. *Journal of Neuroscience Methods*, 297:9–21, 2018. ISSN 0165-0270. doi: https://doi.org/10.1016/j.jneumeth.2017.12.026. URL https://www.sciencedirect.com/science/article/pii/S0165027017304442. [Cited on pages 20 and 21]

[71] P. J. Laub, T. Taimre, and P. K. Pollett. Hawkes processes, 2015. [Cited on page 17]

[72] T. M. Le. A multivariate hawkes process with gaps in observations. *IEEE Transactions on Information Theory*, 64(3):1800–1811, 2018. doi: 10.1109/TIT.2017.2735963. [Cited on page 19]

[73] R. Lemonnier and N. Vayatis. Nonparametric markovian learning of triggering kernels for mutually exciting and mutually inhibiting multivariate Hawkes processes. In T. Calders, F. Esposito, E. Hüllermeier, and R. Meo, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 161–176, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg. [Cited on pages 26 and 27]

[74] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, page 497–506, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605584959. doi: 10.1145/1557019.1557077. URL https://doi.org/10.1145/1557019.1557077. [Cited on page 83]

[75] S. Li, S. Xiao, S. Zhu, N. Du, Y. Xie, and L. Song. Learning temporal point processes via reinforcement learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/5d50d22735a7469266aab23fd8aeb536-Paper.pdf. [Cited on page 21]

[76] S. W. Linderman and R. P. Adams. Discovering latent network structure in point process data. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1413–II–1421. JMLR.org, 2014. URL http://dl.acm.org/citation.cfm?id=3044805.3045050. [Cited on pages 18, 26, 34, 74, and 77]

[77] S. W. Linderman and R. P. Adams. Scalable Bayesian inference for excitatory point process networks. *arXiv preprint arXiv:1507.03228*, 2015. [Cited on pages 3, 18, 26, 34, 74, 77, 78, and 88]

[78] S. W. Linderman, Y. Wang, and D. M. Blei. Bayesian inference for latent Hawkes processes. *NeurIPS Symposium on Advances in Approximate Bayesian Inference Probabilistic*, 2017. [Cited on pages 19 and 27]

[79] S. Liu and M. Hauskrecht. Nonparametric regressive point processes based on conditional gaussian processes. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/9cc138f8dc04cbf16240daa92d8d50e2-Paper.pdf. [Cited on page 21]

[80] G. Loaiza-Ganem, S. Perkins, K. Schroeder, M. Churchland, and J. P. Cunningham. Deep random splines for point process intensity estimation of neural population data. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/d26e5e36c1b0b620407eadabb6c0c5c2-Paper.pdf. [Cited on page 21]

[81] L. Lorch, H. Kremer, W. Trouleau, S. Tsirtsis, A. Szanto, B. Schölkopf, and M. Gomez-Rodriguez. Quantifying the effects of contact tracing, testing, and containment measures in the presence of infection hotspots, 2020. [Cited on page 19]

[82] E. Lukacs. *Characteristic functions*. Griffin, 1970. [Cited on page 15]

[83] C. Mavroforakis, I. Valera, and M. Gomez-Rodriguez. Modeling the dynamics of learning activity on the web. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 1421–1430, Republic and Canton of Geneva, CHE, 2017. International World Wide Web Conferences Steering Committee. ISBN 9781450349130. doi: 10.1145/3038912.3052669. URL https://doi.org/10.1145/3038912.3052669. [Cited on page 20]

[84] H. Mei and J. Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6757–6767, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. [Cited on page 21]

[85] H. Mei, G. Qin, and J. Eisner. Imputing missing events in continuous-time event streams. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4475–4485. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/mei19a.html. [Cited on page 19]

[86] S. Meyer. Self-Exciting Point Processes: Infections and Implementations. *Statistical Science*, 33(3):327 – 329, 2018. doi: 10.1214/18-STS653. URL https://doi.org/10.1214/18-STS653. [Cited on page 19]

[87] S. Meyer and L. Held. Power-law models for infectious disease spread. *The Annals of Applied Statistics*, 8(3):1612–1639, Sept. 2014. doi: 10.1214/14-aoas743. URL https://doi.org/10.1214/14-aoas743. [Cited on page 19]

[88] S. Meyer, J. Elias, and M. Höhle. A space-time conditional intensity model for invasive meningococcal disease occurrence. *Biometrics*, 68(2):607–616, Oct. 2011. doi: 10.1111/j.1541-0420.2011.01684.x. URL https://doi.org/10.1111/j.1541-0420.2011.01684.x. [Cited on page 19]

[89] G. Mohler, F. Schoenberg, M. B. Short, and D. Sledge. Analyzing the world-wide impact of public health interventions on the transmission dynamics of covid-19. *arXiv preprint arXiv:2004.01714*, 2020. [Cited on page 88]

[90] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita. Self-exciting point process modeling of crime. *Journal of the American Statistical Association*, 106(493):100–108, 2011. [Cited on page 21]

[91] A. Morabia. Epidemiology's 350th anniversary. *Epidemiology*, 24(2):179–183, Mar. 2013. doi: 10.1097/ede.0b013e31827b5359. URL https://doi.org/10.1097/ede.0b013e31827b5359. [Cited on page 1]

[92] F. Musmeci and D. Vere-Jones. A space-time clustering model for historical earthquakes. *Annals of the Institute of Statistical Mathematics*, 44(1):1–11, Mar.

1992. doi: 10.1007/bf00048666. URL https://doi.org/10.1007/bf00048666. [Cited on page 19]

[93] Y. Ogata. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, June 1998. doi: 10.1023/a:1003403601725. URL https://doi.org/10.1023/a:1003403601725. [Cited on page 19]

[94] Y. Ogata. On lewis' simulation method for point processes. *IEEE Trans. Inf. Theor.*, 27(1):23–31, Sept. 2006. ISSN 0018-9448. doi: 10.1109/TIT.1981.1056305. URL http://dx.doi.org/10.1109/TIT.1981.1056305. [Cited on page 50]

[95] M. Okatan, M. A. Wilson, and E. N. Brown. Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity. *Neural Computation*, 17(9):1927–1961, Sept. 2005. doi: 10.1162/0899766054322973. URL https://doi.org/10.1162/0899766054322973. [Cited on pages 2 and 20]

[96] T. Omi, n. ueda, and K. Aihara. Fully neural network based model for general temporal point processes. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/39e4973ba3321b80f37d9b55f63ed8b8-Paper.pdf. [Cited on page 21]

[97] T. Ozaki. Maximum likelihood estimation of Hawkes' self-exciting point processes. *Annals of the Institute of Statistical Mathematics*, 31(1):145–155, 1979. [Cited on pages 17 and 58]

[98] A. Paranjape, A. R. Benson, and J. Leskovec. Motifs in temporal networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, page 601–610, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450346757. doi: 10.1145/3018661.3018731. URL https://doi.org/10.1145/3018661.3018731. [Cited on page 83]

[99] C. J. Quinn, T. P. Coleman, N. Kiyavash, and N. G. Hatsopoulos. Estimating the directed information to infer causal relationships in ensemble neural spike train recordings. *Journal of computational neuroscience*, 30(1):17–44, 2011. [Cited on page 53]

[100] J. G. Rasmussen. Bayesian inference for hawkes processes. *Methodology and Computing in Applied Probability*, 15(3):623–642, Sep 2013. ISSN 1573-7713. doi: 10.1007/s11009-011-9272-5. [Cited on pages 6, 18, 43, and 46]

[101] J. G. Rasmussen. Lecture notes: Temporal point processes and the conditional intensity function, 2018. [Cited on page 17]

[102] A. Reinhart. Point process modeling with spatiotemporal covariates for predicting crime, Oct 2018. [Cited on page 21]

[103] P. Reynaud-Bouret, S. Schbath, et al. Adaptive estimation for Hawkes processes; application to genome analysis. *The Annals of Statistics*, 38(5):2781–2822, 2010. [Cited on page 19]

[104] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, 2014. [Cited on page 31]

[105] M.-A. Rizoiu, S. Mishra, Q. Kong, M. Carman, and L. Xie. SIR-hawkes. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18*. ACM Press, 2018. doi: 10.1145/3178876.3186108. URL `https://doi.org/10.1145/3178876.3186108`. [Cited on page 19]

[106] M. G. Rodriguez, J. Leskovec, D. Balduzzi, and B. Schölkopf. Uncovering the structure and temporal dynamics of information propagation. *Network Science*, 2 (1):26–65, 2014. [Cited on page 84]

[107] F. Salehi, W. Trouleau, M. Grossglauser, and P. Thiran. Learning hawkes processes from a handful of events. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 12715–12725. Curran Associates, Inc., 2019. [Cited on pages 25, 58, 69, and 82]

[108] F. P. Schoenberg, M. Hoffmann, and R. J. Harrigan. A recursive point process model for infectious diseases. *Annals of the Institute of Statistical Mathematics*, 71(5):1271–1287, Oct. 2018. doi: 10.1007/s10463-018-0690-9. URL `https://doi.org/10.1007/s10463-018-0690-9`. [Cited on page 19]

[109] O. Shchur, M. Biloš, and S. Günnemann. Intensity-free learning of temporal point processes. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=HygOjhEYDH`. [Cited on page 21]

[110] O. Shchur, N. Gao, M. Biloš, and S. Günnemann. Fast and flexible temporal point processes with triangular maps. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 73–84. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/00ac8ed3b4327bdd4ebbebcb2ba10a00-Paper.pdf`. [Cited on page 21]

[111] C. R. Shelton, Z. Qin, and C. Shetty. Hawkes process inference with missing data. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. [Cited on pages 18, 19, 41, 43, and 58]

[112] M. B. Short, M. R. D'Orsogna, P. J. Brantingham, and G. E. Tita. Measuring and modeling repeat and near-repeat burglary effects. *Journal of Quantitative Criminology*, 25(3):325–339, May 2009. doi: 10.1007/s10940-009-9068-8. URL `https://doi.org/10.1007/s10940-009-9068-8`. [Cited on page 21]

[113] A. Simma and M. I. Jordan. Modeling events with cascades of poisson processes. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, UAI'10, page 546–555, Arlington, Virginia, USA, 2010. AUAI Press. ISBN 9780974903965. [Cited on pages 18 and 77]

[114] J. Snoek, H. Larochelle, and R. P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012. [Cited on page 29]

[115] A. Stomakhin, M. B. Short, and A. L. Bertozzi. Reconstruction of missing data in social networks based on temporal patterns of interactions. *Inverse Problems*, 27(11):115013, oct 2011. doi: 10.1088/0266-5611/27/11/115013. URL `https://doi.org/10.1088/0266-5611/27/11/115013`. [Cited on page 21]

[116] W. Trouleau, J. Etesami, M. Grossglauser, N. Kiyavash, and P. Thiran. Learning hawkes processes under synchronization noise. In *International Conference on Machine Learning*, pages 6325–6334, 2019. [Cited on pages 18, 41, 58, 67, and 74]

[117] W. Trouleau, J. Etesami, M. Grossglauser, N. Kiyavash, and P. Thiran. Cumulants of hawkes processes are robust to observation noise. In *International Conference on Machine Learning*, 2021. [Cited on page 57]

[118] W. Truccolo. From point process observations to collective neural dynamics: Nonlinear hawkes process glms, low-dimensional dynamics and coarse graining. *Journal of Physiology-Paris*, 110(4, Part A):336–347, 2016. ISSN 0928-4257. doi: https://doi.org/10.1016/j.jphysparis.2017.02.004. URL `https://www.sciencedirect.com/science/article/pii/S0928425717300086`. SI: GDR Multielectrode. [Cited on page 20]

[119] U. Upadhyay, A. De, and M. Gomez Rodriguez. Deep reinforcement learning of marked temporal point processes. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL `https://proceedings.neurips.cc/paper/2018/file/71a58e8cb75904f24cde464161c3e766-Paper.pdf`. [Cited on page 21]

[120] P. O. S. Vaz de Melo, C. Faloutsos, R. Assunção, and A. Loureiro. The self-feeding process: a unifying model for communication dynamics in the web. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1319–1330, 2013. [Cited on pages 73 and 75]

[121] P. O. S. Vaz de Melo, C. Faloutsos, R. Assunção, R. Alves, and A. A. Loureiro. Universal and distinct properties of communication dynamics: How to generate realistic inter-event times. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–31, 2015. [Cited on pages 9, 11, 73, 74, and 75]

[122] A. Veen and F. P. Schoenberg. Estimation of space–time branching process models in seismology using an em–type algorithm. *Journal of the American Statistical Association*, 103(482):614–624, 2008. doi: 10.1198/016214508000000148. URL `https://doi.org/10.1198/016214508000000148`. [Cited on pages 18 and 21]

[123] C. Wang and D. M. Blei. Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14(Apr):1005–1031, 2013. [Cited on page 78]

[124] Y. Wang, B. Xie, N. Du, and L. Song. Isotonic hawkes processes. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2226–2234, New York, New York, USA, 20–22 Jun 2016. PMLR. URL `http://proceedings.mlr.press/v48/wangg16.html`. [Cited on page 21]

[125] H. Wold. On stationary point processes and markov chains. *Scandinavian Actuarial Journal*, 1948(1-2):229–240, 1948. [Cited on pages 8, 73, and 74]

[126] W. Wu and N. G. Hatsopoulos. Evidence against a single coordinate system representation in the motor cortex. *Experimental Brain Research*, 175:197–210, 2006. [Cited on pages 51 and 53]

[127] S. Xiao, M. Farajtabar, X. Ye, J. Yan, L. Song, and H. Zha. Wasserstein learning of deep generative point process models. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3250–3259, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. [Cited on page 21]

[128] S. Xiao, J. Yan, M. Farajtabar, L. Song, X. Yang, and H. Zha. Joint modeling of event sequence and time series with attentional twin recurrent neural networks, 2017. [Cited on page 21]

[129] S. Xiao, J. Yan, M. Farajtabar, L. Song, X. Yang, and H. Zha. Learning time series associated event sequences with recurrent point process networks. *IEEE Transactions on Neural Networks and Learning Systems*, 30(10):3124–3136, Oct. 2019. doi: 10.1109/tnnls.2018.2889776. URL `https://doi.org/10.1109/tnnls.2018.2889776`. [Cited on pages 21 and 89]

[130] H. Xu, M. Farajtabar, and H. Zha. Learning Granger causality for Hawkes processes. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1717–1726. PMLR, 2016. URL `http://proceedings.mlr.press/v48/xuc16.html`. [Cited on pages 14, 18, 26, 27, 28, 33, 34, 35, 58, and 82]

[131] H. Xu, D. Luo, and H. Zha. Learning Hawkes processes from short doubly-censored event sequences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 3831–3840. JMLR.org, 2017. URL `http://dl.acm.org/citation.cfm?id=3305890.3306077`. [Cited on pages 19, 25, 41, and 58]

[132] J. Yan, C. Zhang, H. Zha, M. Gong, C. Sun, J. Huang, S. Chu, and X. Yang. On machine learning towards predictive sales pipeline analytics. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 1945–1951. AAAI Press, 2015. ISBN 0262511290. [Cited on pages 18, 43, and 46]

[133] Y. Yang, J. Etesami, N. He, and N. Kiyavash. Online learning for multivariate Hawkes processes. *Neural Information Processing Systems*, 2017. [Cited on pages 18, 55, 58, and 74]

[134] B. Yuan, H. Li, A. L. Bertozzi, P. J. Brantingham, and M. A. Porter. Multivariate spatiotemporal hawkes processes and network reconstruction. *SIAM Journal on Mathematics of Data Science*, 1(2):356–382, 2019. doi: 10.1137/18M1226993. URL `https://doi.org/10.1137/18M1226993`. [Cited on page 21]

[135] B. Yuan, F. P. Schoenberg, and A. L. Bertozzi. Fast estimation of multivariate spatiotemporal hawkes processes and network reconstruction. *Annals of the Institute of Statistical Mathematics*, Jan. 2021. doi: 10.1007/s10463-020-00780-1. URL `https://doi.org/10.1007/s10463-020-00780-1`. [Cited on page 21]

[136] A. Zarezade, A. De, H. Rabiee, and M. Gomez-Rodriguez. Cheshire: An online algorithm for activity maximization in social networks. In *55th Annual Allerton Conference on Communication, Control, and Computing*, 2017. [Cited on page 20]

[137] A. Zarezade, U. Upadhyay, H. Rabiee, and M. Gomez-Rodriguez. Redqueen: An online algorithm for smart broadcasting in social networks. In *WSDM '17: Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, 2017. [Cited on page 20]

[138] C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 2018. [Cited on pages 27 and 30]

[139] Q. Zhang, A. Lipani, O. Kirnap, and E. Yilmaz. Self-attentive Hawkes process. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11183–11193. PMLR, 13–18 Jul 2020. URL `http://proceedings.mlr.press/v119/zhang20q.html`. [Cited on page 21]

[140] W. Zhang, T. K. Panum, S. Jha, P. Chalasani, and D. Page. CAUSE: learning Granger causality from event sequences using attribution methods. *CoRR*,

abs/2002.07906, 2020. URL https://arxiv.org/abs/2002.07906. [Cited on pages 21 and 89]

[141] K. Zhou, H. Zha, and L. Song. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *AISTATS*, volume 31 of *JMLR Workshop and Conference Proceedings*, pages 641–649. JMLR.org, 2013. [Cited on pages 18, 26, 28, 33, 34, 43, 46, and 50]

[142] K. Zhou, H. Zha, and L. Song. Learning triggering kernels for multi-dimensional Hawkes processes. In *International Conference on Machine Learning*, volume 28, pages 1301–1309, 2013. [Cited on pages 18, 26, 28, 35, 55, 58, 66, 69, and 74]

[143] J. Zhuang, Y. Ogata, and D. Vere-Jones. Stochastic declustering of space-time earthquake occurrences. *Journal of the American Statistical Association*, 97(458): 369–380, 2002. doi: 10.1198/016214502760046925. URL https://doi.org/10.1198/016214502760046925. [Cited on page 19]

[144] J. Zhuang, Y. Ogata, and D. Vere-Jones. Analyzing earthquake clustering features by using stochastic reconstruction. *Journal of Geophysical Research: Solid Earth*, 109(B5), 2004. doi: https://doi.org/10.1029/2003JB002879. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2003JB002879. [Cited on page 19]

William **TROULEAU**

Avenue d'Ouchy 24C
1006 Lausanne
Switzerland

linkedin.com/in/william-trouleau
william.trouleau@gmail.com
https://trouleau.github.io
+41 (0)78 814 96 55

French citizen
Swiss C Permit
Born 28.03.1991

KEY COMPETENCES: **Machine Learning • Probabilistic Modeling • Data Mining**

EDUCATION

2015 – 2021 **Ph.D. in Machine Learning** – École Polytechnique Fédérale de Lausanne (EPFL)
- Information & Network Dynamics Lab (INDY), Prof. Matthias Grossglauser and Prof. Patrick Thiran
- Focus on the statistical and algorithmic aspects of **modeling**, **control** and **inference** of **networks** of **times series**; with applications in epidemiology, neuroscience, information diffusion and recommendation systems

2009 – 2015 **B.Sc. and M.Sc. in Communication Systems** – École Polytechnique Fédérale de Lausanne (EPFL)
- Master thesis on user behavior modeling in video-on-demand services (published at KDD 2016)

PROFESSIONAL EXPERIENCE

2019 (3 months) **Research Intern** – Institute for Disease Modeling (Seattle, USA)
- Participated in the *2019 KIT Tuberculosis (TB) Hackathon* to quantify the TB burden in Pakistan
- Took care of data collection, data cleaning, model design and performance evaluation
- Won the competition (out of 9 international teams)

2014 – 2015 (10 months) **Research Intern** – Technicolor (Los Altos, CA, USA)
- Designed a novel generative mixture model that presents a first-of-its-kind characterization of viewer binge-watching behavior on video-on-demand services
- Cleaned & processed >200GB of raw user logs into an accessible MongoDB database, kickstarting several projects for my fellow researchers

2013 (3 months) **Research Intern** – Technicolor (Paris, France)
- Designed a hierarchical topic model with Monte Carlo Markov Chain inference for legal document classification and exploratory analysis of patent portfolios
- Presented the results to both the research and legal teams

TEACHING AND OUTREACH EXPERIENCE

2015-2020 **Teaching Assistant** – EPFL
- Led the rapid conversion of 2 courses to an online format due to Covid-19 restrictions at EPFL
- Taught a number of lectures for several courses, including the cornerstone "Stochastic Models for Communications" course for undergraduate students at EPFL (~100 students)
- Designed data mining exercises in Python/Spark. Recruited and led several teams of teaching assistants for an undergraduate data mining course "Internet Analytics" (~50 students) Received Teaching Assistant Award for my work
- Oversaw exercise sessions and evaluated exams for undergraduate courses each semester

2015-2017 **Initial Study Advisor** – EPFL
- Helped the Deputy of Section with new M.Sc. students in "Communication Systems"
- Mentored >50 students over 3 years

SKILLS

Data Science Machine Learning, Optimization, Probabilistic Modeling, Applied Data Analysis, Information Theory, Statistical Signal Processing

Programming Python, Spark, R, Matlab, Bash, Java, SQL/NoSQL, HTML/CSS, LaTeX

LANGUAGES AND MISCELLANEA

Languages
- **French:** Native language
- **English:** Fluent, written and spoken

Services
- Reviewer for machine learning conferences: ICML, NeurIPS, and AISTATS
- Initial study advisor for new EPFL M.Sc. students

HONORS AND AWARDS

2020 In Top 33% of Reviewers, ICML'20
2019 Hackathon Winner, *2019 Hack TB, KIT*
2019 Teaching Assistant Award, EPFL
2016 Student Travel Award, ACM SIGKDD
2015 EDIC Fellowship, EPFL

SELECTED PUBLICATIONS

2021 **Cumulants of Hawkes Processes are Robust to Observation Noise**
W. Trouleau, J. Etesami*, N. Kiyavash, M. Grossglauser, P. Thiran.* ***ICML 2021****.*

2021 **A Variational Inference Approach to Learning Multivariate Wold Processes**
J. Etesami, W. Trouleau*, N. Kiyavash, M. Grossglauser, P. Thiran.* ***AISTATS 2021****.*

2020 **Quantifying the Effects of Contact Tracing, Testing, and Containment Measures in the Presence of Infection Hotspots.**
L. Lorch, H. Kremer, W. Trouleau, S. Tsirtsis, A. Szanto, B. Schölkopf, M. Gomez-Rodriguez. Preprint.

2019 **Learning Hawkes Processes Under Synchronization Noise**
W. Trouleau, J. Etesami, M. Grossglauser, N. Kiyavash, P. Thiran. ***ICML 2019****.*

2019 **Learning Hawkes Processes from a Handful of Events**
F. Salehi, W. Trouleau*, M. Grossglauser, P. Thiran.* ***NeurIPS 2019****.*

2018 **Stochastic Optimal Control of Epidemic Processes in Networks**
L. Lorch, A. De, S. Bhatt, W. Trouleau, U. Upadhyay, M. Gomez-Rodriguez. Machine Learning for Health Workshop at NeurIPS, 2018. ***ML4H 2018****.*

2017 **Prev. chemotherapy to control soil-transmitted helminthiasis averted more than 500'000 DALYs in 2015**
A. Montresor, W. Trouleau, D. Mupfasoni, M. Bangert, S.A. Joseph, A. Mikhailov, C. Fitzpatrick. **Transactions of The Royal Society of Tropical Medicine and Hygiene**, 2017.

2016 **Just one more: Modeling binge watching behavior**
W Trouleau, A Ashkan, W Ding, B Eriksson. ***KDD 2016****.*

** Equal contribution*