Algorithms and Inference for Simultaneous-Event Multivariate Point-Process, with Applications to Neural Data

by

Demba Ba

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2011

Author ....
Department of Electrical Engineering and Computer Science
May 19, 2011

Certified .....
Emery N. Brown
Professor of Computational Neuroscience
and Professor of Health Sciences and Technology.
Thesis Supervisor

Accepted by ....
Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Students

Algorithms and Inference for Simultaneous-Event Multivariate Point-Process, with Applications to Neural Data

by

Demba Ba

Submitted to the Department of Electrical Engineering and Computer Science
on May 19, 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering

Abstract

The formulation of multivariate point-process (MPP) models based on the Jacod likelihood does not allow for simultaneous occurrence of events at an arbitrarily small time resolution. In this thesis, we introduce two versatile representations of a simultaneous-event multivariate point-process (SEMPP) model to correct this important limitation. The first one maps an SEMPP into a higher-dimensional multivariate point-process with no simultaneities, and is accordingly termed the *disjoint* representation. The second one is a marked point-process representation of an SEMPP, which leads to new thinning and time-rescaling algorithms for simulating an SEMPP stochastic process. Starting from the likelihood of a discrete-time form of the disjoint representation, we present derivations of the continuous likelihoods of the disjoint and MkPP representations of SEMPPs.

For static inference, we propose a parametrization of the likelihood of the disjoint representation in discrete-time which gives a multinomial generalized linear model (mGLM) algorithm for model fitting. For dynamic inference, we derive generalizations of point-process adaptive filters. The MPP time-rescaling theorem can be used to assess model goodness-of-fit.

We illustrate the features of our SEMPP model by simulating SEMPP data and by analyzing neural spiking activity from pairs of simultaneously-recorded rat thalamic neurons stimulated by periodic whisker deflections. The SEMPP model demonstrates a strong effect of whisker motion on simultaneous spiking activity at the one millisecond time scale. Together, the MkPP representation of the SEMPP model, the mGLM and the MPP time-rescaling theorem offer a theoretically sound, practical tool for measuring joint spiking propensity in a neuronal ensemble.

Thesis Supervisor: Emery N. Brown
Title: Professor of Computational Neuroscience
and Professor of Health Sciences and Technology.

Acknowledgments

"How does it feel?", "What's next?". Obviously, many have asked me these questions. This is not the appropriate place to answer the second question. However, I will try to answer the second one in a few sentences.

I have mixed feelings about my experience as a graduate student. I, as the majority of grad students, had to battle with many of the systemic idiosyncrasies of grad school and academia. A shift occurred in my way of thinking when it became obvious that the said idiosyncrasies were taking a toll on my experience as a graduate student. I told myself that I would make my PhD experience what I wanted to be. That is precisely what I did after my M.S.I thought about the best way of satisfying graduation requirements while building an academic profile that reflected my own view of science/knowledge/research. I followed that approach and, simply because of this, I am happy with wherever it has led me, as a thinker, as an academic, as a researcher.

A number of people have helped me to achieve this objective. First, I would like to thank my thesis supervisor Emery Brown, who picked me up at a time when I was having a difficult time transitioning from M.S. to PhD. For his moral support and trust, and a number of other reasons, I am forever grateful. I would also like to thank George Verghese who has consistently given me moral support since the 2004 visit day at MIT when he saw me in a corner and said "Don't be shy, you should mingle and talk to people...". I would also like to thank professor Terry Orlando for his moral and financial support during the transition period between M.S. and PhD. Many thanks to professor John Tsitsiklis for agreeing to serve on my thesis committee, as well as professor Sanjoy Mitter.

I would like to thank my family for their support during these arduous years. My mom Fama, my dad Bocar, my sister and brothers: Famani, Khalidou, Moussa and Moctar. The latter two's sons and daughters also deserve mention here: let's talk if and when you have to decide between Harvard and MIT. I would also like to thank my cousin Elimane Kamara, my aunt Gogo Aissata, Rose, my aunt Tata Rouguy and

her husband Tonton Hamat.

I would like to thank my friends. My first contact with MIT was through Zahi Karam, who told me to go to Georgia Tech instead. (Un)fortunately, Tech didn't accept me. Alaa Kharbouch, Hanan Karam, Akram Sadek, Mack Durham, Najim Dehak, Yusef Asaf, Valerie Loehr, Yusef Mestari, thanks for the good times and the moral support. Borjan Gagoski is the first of my cohorts that I met: we've been through the tough years together. Among cohorts, I'd also like to thank Faisal Kashif, Bill Richoux and Jennifer Roberts (correction, Jen and I went to University of Maryland College Park together...sorry Borjan). Also at MIT, I would like to thank Sourav Dey and his wife (Pallabi), Obrad and Danilo Scepanovic, Miriam Makhlouf (This Is Africa, TIA, represent!), Hoda Eydgahi, Camilo Lamus, Antonio Molins, Ammar Ammar, Yi Cai, Paul Azunre, Paul Njoroge (African man!) and Ali Motamedi. I would also like to thank Siragan Gailus, Sangmin Oh, Sangyun Kim, Nevin Raj, Subok Lee, Caroline Lahoud, Antonio Marzo and Shivani Rao.

Through my time at MIT, I have interacted with certain members of the MIT staff, who I cannot forget to acknowledge: Lourenco Pires, Debb of MTL, Lisa Bella, Sukru Cinar and Pierre of custodial services.

Special thanks to the people who helped me to get industry experience at MSR and Google, and coached me through those experiences: Rico Malvar, Dinei Florencio, Phil Chou (all three of MSR), Kevin Yu and Xinyi Zhang, both of Google. It has been a pleasure interacting with you and meeting you. I have also made friends through these internships, notably Flavio Ribeiro at MSR, Lauro Costa and Shivani Rao at Google.

Many thanks to my friends from back home: Lahad Fall, Fatou Diagne, Suzane Diop, Wahab Diop and their family, Pape Sylla, Babacar Diop and Amadou Racine Ly. I would also like to thank my high-school friend Said Pitroipa.

I would like to thank professors Staffilani and Melrose of the MIT math department for being excellent teachers, and inspiring me.

Last but not least, I would like to thank current and ex members of the Brown Lab for bearing with me, notably Sage, Hideaki, Neal and Deriba.

Contents

List of Figures

List of Tables

Chapter 1

Background, Introduction and Scope

1.1 Introduction

Neuroscientists explore how the brain works by applying sensory stimuli and recording the responses of neurons. Their goal is to understand the respective contributions of the stimulus, as opposed to the neurons' intrinsic dynamics, to the observed activity. Nowadays, it is not uncommon to record simultaneously from multiple neurons. However, techniques for the sound analysis of data generated by such experiments have been lagging a step behind.

Motivated mainly by applications in neuroscience, the aim of this thesis is to develop a generic framework for the rigorous analysis of multivariate-point-process phenomena. Perhaps it is easier to understand the scope of this thesis by dissecting its title. Loosely, a uni-variate *point process* is a sequence of discrete events (e.g. firing of a neuron, arrival of a bus/passenger at a station) that occur at random points in continuous time (or space). In general, there could be multiple such processes evolving in parallel, in which case we speak of a *multivariate* point-process. Effectively, a multivariate point process is a finite-dimensional vector process, the components of which are uni-variate point processes. In the literature, significant attention has been given to the particular case of multivariate point processes for which the probability of *simultaneous* events/arrivals in any pair of components is negligible. Meanwhile, the case where simultaneous arrivals/events in multiple components cannot be ignored has received little to no attention. From a theoretical standpoint, this latter fact is the main motivation of this thesis.

For data analysis tasks, this thesis develops *algorithms* for simulation and estimation of simultaneous-event multivariate point-process models. In practice, the type of *inference*/estimation problems we would like to solve fall within the class of parametric density estimation problems. We call such problems 'static' if the parameters of the model are fixed. If one allows the parameters of the model to vary (e.g. with time), then we call such models 'dynamic'. We demonstrate the efficacy of the inference and simulation algorithms on *neural data.* These data consist of spiking activity from simultaneously-recorded rat thalamic neurons stimulated by periodic whisker deflections.

1.2 Background

Existing techniques to analyze neural data fall mainly into two categories: likelihood methods and those that do not make strong assumptions, if any, about the generating process of the data. We term the latter non-likelihood methods.

1.2.1 Non-likelihood methods

As a class, non-likelihood based methods are limited due to their inability to quantify the extent to which the stimulus, as opposed to spiking history, modulates the joint activity of a group of neurons. The cross-correlogram and the cross-intensity function are two similar approaches which reduce the problem of analyzing ensemble neural data to one of characterizing the relationships between pairs of neurons. Given a pair of neural spike trains and a fixed bin width, the un-normalized cross-correlogram [8] is the deterministic cross-covariance between the two spike trains, computed at a series of lags. An underlying assumption of this method is that of stationarity, which loosely states that the joint statistics of the pair of neurons do not change over time. Although convenient, such an assumption is hard to justify given how plastic neural systems are. The cross-intensity function [6] estimates the rate of a given neuron at different lags relative to another neuron. In spite of its simplicity, the cross-intensity function has not gained as much popularity as the cross-correlogram within

the neuroscience community. The joint peri-stimulus time histogram (JPSTH) [17] is another histogram-based method which operates on pairs of neurons. The JPSTH is the natural extension to pairs of neurons of the well-known PSTH: it is a two-dimensional histogram displaying the joint spike count per unit time at each time u for the first neuron and time v for the second neuron. The JPSTH addresses one of the drawbacks of the cross-correlogram, which is the stationarity assumption within trials. However, due to its reliance on a stationarity assumption across trials, the JPSTH may lead to incorrect conclusions when there exists across-trial dynamics. In [45], the authors incorporate a statistical model for time-varying joint-spiking activity within the JPSTH framework. They show that this allows for more efficient computation of the joint-firing rate of pairs of neurons. Among non-likelihood methods, spike pattern classification techniques allow one to analyze associations beyond pairwise ones. These methods can be used to assess the statistical significance of certain spike patterns among multiple neurons [1, 18, 19, 34]. One of the challenges posed by spike pattern classification is that of selecting the appropriate pattern size.

1.2.2 Likelihood methods

Likelihood methods are closest in spirit to the framework we propose in this thesis. Among such methods, there are those based on information geometry [3, 31] and those based on point processes [24, 33]. Likelihood methods based on information geometry rely on an expansion of the log of the joint pmf of a vector binary process as a linear combination of its moments. Recently, a method was proposed which combines information geometry and adaptive filtering to track the evolution over time of the moments of a vector binary process [39]. The nature of experiments in neuroscience is such that it is natural to expect the joint statistics of single neurons or of an ensemble to vary with time. In [39], the authors use a stochastic continuity constraint on the moments in order to recover the time-varying nature of the statistics of the data. One would expect, however, that the stimulus and/or spiking history of neurons in an ensemble would encode information about the time-varying nature of the joint statistics of the ensemble. This is precisely what point-process methods

attempt to do. Building on their success in characterizing single-neuron data [43], point-process methods have been shown to provide a sensible framework within which one is able to isolate the contributions of stimulus as opposed to history to the joint activity of a group of neurons [33]. However, there is a caveat: the assumption that, for small enough time resolution, the probability of joint firing among any two or more neurons in the ensemble is negligible. As the time resolution becomes arbitrarily small, this leads to the Jacod likelihood for multivariate point processes with no simultaneities [22, 14]. The Jacod likelihood is expressed as the product of univariate point-process likelihoods.

1.3 Contributions

Likelihood methods based on point processes assume that either the components of the multivariate point process are independent, or that simultaneous occurrences of events in any two components can be neglected. These assumptions turn out to be convenient as, in both cases, one can fit an approximate model to the multivariate point process by performing inference separately on each of its components. The case where the probability of simultaneous occurrences cannot be neglected has received little to no attention in the literature. Ventura et al. [45] developed a likelihood procedure to overcome this limitation for analyzing a pair of neurons. In [25], Kass et al. extend Ventura's approach to multiple neurons. Solo [41] recently reported a simultaneous event multivariate point-process (SEMPP) model to correct this important limitation. However, in his treatment, Solo does not provide a framework for inference based on real data. Here, we propose a quite general framework for inference based on SEMPP observations. We introduce two representations of an SEMPP. The so-called *disjoint* representation transforms an SEMPP into an auxiliary multivariate point-process with no simultaneities. The multivariate point-process theorem [14] can be applied to this new representation to assess model goodness-of-fit. The marked point-process (MkPP) representation [14] leads to algorithms for simulating an SEMPP stochastic process. In discrete-time (DT), the likelihood of

the disjoint representation can be expressed as a product of conditional multinomial trials (rolls of a dice). Starting from such an approximation, we derive the limiting continuous-time likelihood, i.e. that of the continuous-time (CT) disjoint representation. We also derive a form of this likelihood in terms of the MkPP representation. In practice, model fitting is performed in discrete-time. We propose a parametrization of the likelihood of the disjoint process in discrete-time which turns it into a multivariate generalized linear model (mGLM) with multinomial observations and logit link [16]. We propose and make available a very efficient implementation of the mGLM, which is up to an order of magnitude faster than standard implementations, such as Matlab's. Last but not least, we derive natural generalizations of point-process adaptive filters that are able to handle simultaneous occurrences of events in multivariate point processes.

We apply our methods to the analysis of data recorded from pairs of neurons in the rat thalamus in response to periodic whisker deflections varying in velocity. Our model provides a direct estimate of the magnitude of simultaneous spiking propensity and the degree to which whisker stimulation modulates this propensity.

Chapter 2

Discrete-Time and Continuous-Time Likelihoods of SEMPPs

In this chapter, we begin with a simple definition of an SEMPP. Then, We show an explicit one-to-one mapping of an SEMPP to an auxiliary MPP with no simualteneities, albeit in a higher dimensional space. We call this new MPP the *disjoint* representation. The disjoint representation admits an alternate equivalent representation as an MkPP with finite mark space, which we also develop here. Last, we derive discrete-time and continuous-time SEMPP likelihoods. In discrete-time, the likelihood of the disjoint representation can be expressed as a product of conditional multinomial trials. Starting from this likelihood, we derive the continuous-time likelihood of the disjoint process by taking limits. We also derive a form of the continuous-time likelihood in terms of the MkPP representation. The Jacod and univariate point-process likelihoods are special cases of the continuous-time likelihoods obtained here.

We walk the reader through all key derivations. The less essential derivations are shown in one of the appendices.

2.1 Simultaneous-event Multivariate Point Process

We consider an observation interval $(0, T]$ and, for $t \in (0, T]$, let $N(t) = (N_1(t), N_2(t), \cdots, N_C(t))'$ be a C-variate point-process defined as $N_c(t) = \int_0^t dN_c(u)$, where $dN_c(t)$ is the indicator function which is 1 if there is an event at time t and 0 otherwise, for $c = 1, \cdots, C$. $N_c(t)$ counts the number of events for component c in the interval $(0, t]$. We assume

that each component c has a conditional intensity function (CIF) defined as

$$\lambda_c(t|H_t) = \lim_{\Delta \to 0} \frac{P[N_c(t+\Delta) - N_c(t) = 1|H_t]}{\Delta}, \tag{2.1}$$

where H_t is the history of the C-variate point process up to time t. Let $dN(t) = (dN_1(t), dN_2(t), \cdots, dN_C(t))'$ be the vector of indicator functions $dN_c(t)$ at time t. We may treat $dN(t)$ as a C-bit binary number. Therefore, there are 2^C possible outcomes of $dN(t)$ at any t. C of these outcomes have only one non-zero bit (that is, only one event in one component of $dN(t)$) and $2^C - C - 1$ have two or more non-zero bits. That is, there is an event at time t in at least 2 components of $dN(t)$. The last outcome is $dN(t) = (0, \cdots, 0)'$.

We define $N(t)$ as a *simultaneous-event multivariate point process* (SEMPP) if, at any time t, $dN(t)$ has at least two non-zero bits. That is, events are observed simultaneously in at least two of the components of $N(t)$. The special case in which, at any t, $dN(t)$ can only take as values one of the C outcomes for which only one of the bits of $dN(t)$ is non-zero is the multivariate point process defined by Vere-Jones [14]. The joint probability density of $N(t)$ in this special case is given by the Jacod likelihood function [32], [24, 14].

2.2 The disjoint and marked point-process representations

We introduce the *disjoint* representation, which maps an SEMPP into an auxilliary MPP with no simultaneities, in a higher dimensional space. This new disjoint MPP admits an alternate representation as marked point-process with finite mark space.

2.2.1 The disjoint representation

To derive the joint probability density function of an SEMPP, we develop an alternative representation of $N(t)$. Let $M = 2^C$ be the number of possible outcomes of $dN(t)$ at t. We define a new $M-1$-variate point process $N^*(t) = (N_1^*(t), N_2^*(t), \cdots, N_{M-1}^*(t))'$ of disjoint outcomes of $N(t)$. That is, each component of $N^*(t)$ is a counting process for one and only one of the $2^C - 1$ outcomes of $dN(t)$ (patterns of C bits) that have

at least one non-zero bit. For any t, the vector $dN^*(t) = (dN_1^*(t), \cdots, dN_{M-1}^*(t))'$ is an $M-1$-bit binary number with at most one non-zero bit. The non-zero element of $dN^*(t)$ (if any) is an indicator of the pattern $dN(t)$ of C bits which occurs at t. $dN^*(t) = (0, \cdots, 0)'$ corresponds to $dN(t) = (0, \cdots, 0)'$. We define the CIF of $N_m^*(t)$ as

$$\lambda_m^*(t|H_t) = \lim_{\Delta \to 0} \frac{P[N_m^*(t+\Delta) - N_m^*(t) = 1|H_t]}{\Delta}, \tag{2.2}$$

where the counting process is $N_m^*(t) = \int_0^t dN_m^*(u)$. We term $N^*(t)$ the *disjoint* process or representation.

One simple way to map from $dN(t)$ to $dN^*(t)$ is to treat the former as a C-bit binary number, reverse the order of its bits, and convert the resulting binary number to a decimal number. We use this decimal number as the index of the non-zero component of $dN^*(t)$. The inverse map proceeds by finding the index of the non-zero entry of $dN^*(t)$, expressing this index as a C-bit binary number, and reversing the order of the bits to obtain $dN(t)$. This one-to-one map is described in detail in the next few pages for the arbitrary C-variate case. First, we illustrate this one-to-one map in Table 2.1 for the case $C = 3$ and $M = 8$. In this example, $N(t)$ is related to $N^*(t)$ by

$$N_1(t) = N_1^*(t) + N_3^*(t) + N_5^*(t) + N_7^*(t) \tag{2.3}$$

$$N_2(t) = N_2^*(t) + N_3^*(t) + N_6^*(t) + N_7^*(t) \tag{2.4}$$

$$N_2(t) = N_4^*(t) + N_5^*(t) + N_6^*(t) + N_7^*(t). \tag{2.5}$$

The CIFs of $N(t)$ are related to those of $N^*(t)$ in a similar fashion.

From $N(t)$ to $N^*(t)$: For each $t \in (0, T]$, the vector $dN(t) = (dN_1(t), \cdots, dN_C(t))'$ of counting measure increments of $N(t)$ has entries either 0 or 1. Therefore, we can treat $dN(t)$ as a C-length binary number. We let $m_{dN(t)} = \sum_{c=1}^{C} dN_c(t) 2^{c-1}$ be the decimal (base-10) representation of $dN(t)$: $m_{dN(t)} \in \{0, \cdots, 2^C - 1\}$.

Consider the 2^C-1-dimensional vector $dN^*(t) = (dN_1^*(t), \cdots, dN_{2^C-1}^*(t))'$. If $m_{dN(t)} =$

Table 2.1. Map from $dN(t)$ to $dN^*(t)$, $C = 3$, $M = 8$

$dN(t)'$	m	$dN^*(t)'$
(1,0,0)	1	(1,0,0,0,0,0,0)
(0,1,0)	2	(0,1,0,0,0,0,0)
(1,1,0)	3	(0,0,1,0,0,0,0)
(0,0,1)	4	(0,0,0,1,0,0,0)
(1,0,1)	5	(0,0,0,0,1,0,0)
(0,1,1)	6	(0,0,0,0,0,1,0)
(1,1,1)	7	(0,0,0,0,0,0,1)

0, we let $dN^*(t) = (0, \cdots, 0)'$. Otherwise, we let $dN^*_m(t) = 1$ if $m = m_{dN(t)}$ and $dN_m(t) = 0$ otherwise. In this case, $dN^*(t)$ is an indicator vector for the event $dN(t)$ which occurs at t. If we let $N^*_m(t) = \int_0^t dN^*_m(u)$, then $N^*(t) = (N^*_1(t), \cdots, N^*_{2^C-1}(t))'$ becomes a multivariate point-process of disjoint events from $N(t)$.

From $N^*(t)$ to $N(t)$: For each $t \in (0, T]$, the vector $dN^*(t) = (dN^*_1(t), \cdots, dN^*_{2^C-1}(t))'$ is either $(0, \cdots, 0)'$ or an indicator vector. In the former case, we let $dN(t) = (0, \cdots, 0)'$. In the latter case, we would like to determine the event $dN(t)$ that $dN^*(t)$ is an indicator of. Let $m \in \{1, \cdots, 2^C - 1\}$ be the index of the non-zero entry of $dN^*(t)$ and $b_m = b_{m1}b_{m2}\cdots b_{mC}$ be the binary representation of m. If we let $dN(t) = (b_{mC}, \cdots, b_{m2}, b_{m1})'$, we obtain the event $dN(t)$ that $dN^*(t)$ is an indicator of. Letting $N_c(t) = \int_0^t dN_c(u)$, we recover the C-variate SEMPP $N(t) = (N_1(t), \cdots, N_2(t))'$.

2.2.2 The marked point-process representation

We give the following definition, adapted from [14], of a marked point process on the real line.

Definition: *A marked point process with locations on the real line $\mathbb{R}$ and marks in the complete separable metric space $\mathcal{M}$, is a point process $\{(t_\ell, m_\ell)\}$ on $\mathbb{R} \times \mathcal{M}$ with the additional property that the unmarked process $\{t_\ell\}$ is a point process in its own right, called the ground process and denoted $N_g(\cdot)$.*

When $\mathcal{M}$ is a finite set, we say that the MPP is an MkPP with *finite* mark space. In-

tuitively, one may think of an MkPP as follows: (a) events occur at random points in continuous-time (or space) according to the ground process, (b) every time an event occurs, one assigns a mark to this event by drawing a sample from a distribution which may very well depend on time, as well as the history of the ground process and/or past marks.

If we let $0 < t_1 < t_2 < \cdots < t_L \leq T$ denote the times in the observation interval $(0, T]$ at which $dN(t)$ has at least one non-zero bit, then we can express the disjoint process $N^*(t)$ as a marked point process (MkPP) $\{(t_\ell, dN^*(t_\ell)\}_{\ell=1}^L$ with M-1-dimensional mark space. At t_ℓ, at least one of the bits of $dN(t)$ is non-zero. The unmarked process $\{t_\ell\}_{\ell=1}^L$ is the ground point process [14]. The mark, which is the index m_ℓ of the non-zero bit of $dN^*(t_\ell)$ then indicates, through the map described above, exactly which of the $M-1$ patterns of C bits (outcomes of $dN(t)$ other than $(0, \cdots, 0)'$) occurred at t_ℓ. At any other t, $dN(t) = (0, \cdots, 0)'$.

We denote by $dN_g(t)$ the indicator function that is 1 at t_ℓ, $\ell = 1, \cdots, L$ and zero at any other t. The ground point process defines the times of occurrence of *any* pattern of C bits (outcomes of $dN(t)$) that are not all zero. For each m, the times at which $dN_m^*(t)$ is non-zero define the times of occurrence of *one specific* pattern of C bits that are not all zero. It follows that the counting process and the CIF of the ground point process are respectively

$$N_g(t) = \sum_{m=1}^{M-1} N_m^*(t) \tag{2.6}$$

$$\lambda_g^*(t|H_t) = \sum_{m=1}^{M-1} \lambda_m^*(t|H_t). \tag{2.7}$$

The probability of the marks is given by the multinomial probability mass function

$$P[dN_m^*(t) = 1 | dN_g(t) = 1, H_t] = \frac{\lambda_m^*(t|H_t)}{\lambda_g^*(t|H_t)}, \tag{2.8}$$

for $m = 1, \cdots, M-1$. The derivations for Eqs. 2.7 and 2.8 are in Appendix A. The MkPP representation provides an efficient description of $N(t)$. The probability of an

event occurring in $(0, T]$ is governed by the CIF $\lambda_g^*(t|H_t)$ of the ground point process. When an event is observed in $dN_g(t)$, the marks are drawn from an $M-1$-dimensional history-dependent multinomial distribution (Eq. 2.8) to produce the corresponding event in $N^*(t)$, or equivalently $N(t)$.

N.B: The careful reader will notice that I am being a bit cavalier when using the notation $dN^*(t_\ell)$: this is the indicator vector, the index of the nonzero entry of which is m_ℓ. For any t_ℓ, $\ell = 1, \cdots, L$, $dN^*(t_\ell)$ is automatically an indicator vector. For any other $t \neq t_\ell$, $dN^*(t)$ is the zero vector. So, in short, $dN^*(t_\ell)$ and its non-zero index are two ways of representing the mark. I struggled with how to deal with the notation. In the end, this made the most sense. Hopefully, this does not cause too much confusion.

2.3 Likelihoods

Our goal is to derive the joint probability density function (PDF) of an SEMPP in discrete and continuous-time using straightforward heuristic arguments. We start with the likelihood for a discrete-time form of the disjoint representations and obtain continuous-time likelihoods by taking limits.

2.3.1 Discrete-time likelihood

To derive the joint PDF of $N^*(t)$ in discrete time, we define the discrete-time representations of $N(t)$ and $N^*(t)$.

Choose I large and partition the interval $(0, T]$ into sub-intervals of width $\Delta = I^{-1}T$. In discrete-time $N_c(t)$ and $N_m^*(t)$ are respectively $N_{c,i} = N_c(i\Delta)$, $N_{m,i}^* = N_m^*(i\Delta)$ for $i = 1, \cdots, I$. Let $\Delta N_{c,i} = N_{c,i} - N_{c,i-1}$, and $\Delta N_{m,i}^* = N_{m,i}^* - N_{m,i-1}^*$. Letting $\Delta N_i = (\Delta N_{1,i}, \cdots, \Delta N_{C,i})'$, we choose I large enough so that $\Delta N_{c,i}$ is 0 or 1. Either $\Delta N_i^* = (\Delta N_{1,i}^*, \cdots, \Delta N_{M-1,i}^*)'$ has one event in exactly one component or $\Delta N_i^* = (0, \cdots, 0)'$. Let $\Delta N^* = (\Delta N_1^*, \cdots, \Delta N_I^*)'$ be the $I \times M-1$ matrix of discretized outcomes for the observation interval $(0, T]$. Each ΔN_i^*, where i is the discrete-time index, is a realization from a multinomial trial with M outcomes (roll

of an M-sided die):

$$P\left[\Delta N_i^* | H_i\right] = \prod_{m=1}^{M-1} \left(\lambda_m^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*} \left(1 - \sum_{m=1}^{M-1} \lambda_m^*[i|H_i]\Delta\right)^{1-\sum_{m=1}^{M-1}\Delta N_{m,i}^*}, \quad (2.9)$$

$$= \prod_{m=1}^{M-1} \left(\lambda_m^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}}, \quad (2.10)$$

where $\Delta N_{g,i} = N_{g,i} - N_{g,i-1} = \sum_{m=1}^{M-1} \Delta N_{m,i}^*$, $N_{g,i} = N_g(i\Delta)$. The probability mass function of ΔN^* can be written as the product of conditional M-nomial trial:

$$P\left[\Delta N^*\right] = \prod_{i=1}^{I} P\left[\Delta N_i^* | H_i\right] + o(\Delta^L) \quad (2.11)$$

$$= \prod_{i=1}^{I} \prod_{m=1}^{M-1} \left(\lambda_m^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}} + o(\Delta^L). \quad (2.12)$$

We note that Eq. 2.12 can also be expressed in terms of a discrete-time form of the MkPP representation A.13. The manipulations are detailed in Appendix A.

2.3.2 Continuous-time likelihoods

Disjoint likelihood

We can obtain the continuous-time likelihood $p\left[N_{(0,T]}^*\right]$ of the disjoint process $N^*(t)$ by relating it to the discrete-time likelihood of Eq. 2.12 and then taking limits:

$$P\left[\Delta N^*\right] \approx p\left[N_{[0,T)}^*\right] \Delta^L. \quad (2.13)$$

Therefore,

$$p\left[N_{(0,T]}^*\right] = \lim_{\Delta \to 0} \frac{P\left[\Delta N^*\right]}{\Delta^L}. \quad (2.14)$$

Below, we show that $p\left[N_{(0,T]}^*\right]$ is the product of $M-1$ continuous-time univariate point process likelihoods.

First, we approximate Eq. 2.12 as follows:

$$P[\Delta N^*] = \prod_{i=1}^{I}\prod_{m=1}^{M-1}\left(\frac{\lambda_m^*[i|H_i]\Delta}{1-\lambda_g^*[i|H_i]\Delta}\right)^{\Delta N_{m,i}^*}\left(1-\lambda_g^*[i|H_i]\Delta\right)+o(\Delta^L) \tag{2.15}$$

$$\approx \prod_{i=1}^{I}\prod_{m=1}^{M-1}\left(\lambda_m^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*}\exp\left\{-\lambda_g^*[i|H_i]\Delta\right\}+o(\Delta^L) \tag{2.16}$$

$$= \exp\left\{\sum_{i=1}^{I}\sum_{m=1}^{M-1}\Delta N_{m,i}^*\left(\log\lambda_m^*[i|H_i]\Delta\right)-\lambda_g^*[i|H_i]\Delta\right\}+o(\Delta^L) \tag{2.17}$$

$$= \exp\left\{\sum_{m=1}^{M-1}\sum_{i=1}^{I}\Delta N_{m,i}^*\log\lambda_m^*[i|H_i]\Delta-\lambda_m^*[i|H_i]\Delta\right\}+\ o(\Delta^L), \tag{2.18}$$

where we have substituted $\lambda_g^*[i|H_i]=\sum_{m=1}^{M-1}\lambda_m^*[i|H_i]$. Then, we simplify $P[\Delta N^*]/\Delta^L$ as

$$\frac{P[\Delta N^*]}{\Delta^L} \approx \frac{\exp\left\{\sum_{m=1}^{M-1}\sum_{i=1}^{I}\Delta N_{m,i}^*\log\lambda_m^*[i|H_i]\Delta-\lambda_m^*[i|H_i]\Delta\right\}+o(\Delta^L)}{\Delta^L} \tag{2.19}$$

$$= \frac{\exp\left\{\sum_{m=1}^{M-1}\sum_{i=1}^{I}\Delta N_{m,i}^*\log\lambda_m^*[i|H_i]-\lambda_m^*[i|H_i]\Delta\right\}\Delta^L+o(\Delta^L)}{\Delta^L} \tag{2.20}$$

$$= \prod_{m=1}^{M-1}\exp\left\{\sum_{i=1}^{I}\Delta N_{m,i}^*\left(\log\lambda_m^*[i|H_i]\right)-\sum_{i=1}^{I}\lambda_m^*[i|H_i]\Delta\right\}+\frac{o(\Delta^L)}{\Delta^L}. \tag{2.21}$$

Finally, we can obtain $p[N_{[0,T)}^*]$ by passing to the limit:

$$p[N_{[0,T)}^*] = \lim_{\Delta\to 0}\prod_{m=1}^{M-1}\exp\left\{\sum_{i=1}^{I}\Delta N_{m,i}^*\left(\log\lambda_m^*[i|H_i]\right)-\sum_{i=1}^{I}\lambda_m^*[i|H_i]\Delta\right\}+\frac{o(\Delta^L)}{\Delta^L} \tag{2.22}$$

$$= \prod_{m=1}^{M-1}\lim_{\Delta\to 0}\exp\left\{\sum_{i=1}^{I}\Delta N_{m,i}^*\left(\log\lambda_m^*[i|H_i]\right)-\sum_{i=1}^{I}\lambda_m^*[i|H_i]\Delta\right\} \tag{2.23}$$

$$= \prod_{m=1}^{M-1}\exp\left\{\int_0^T\log\lambda_m^*(t|H_t)dN_m^*(t)-\int_0^T\lambda_m^*(t|H_t)dt\right\}. \tag{2.24}$$

If we let $N^*(t)$ be the multivariate point process defined by restricting $dN^*(t)$ to the C components which are indicators for the outcomes for which only one bit of $dN(t)$ is non-zero (that is, if we disregard simultaneous occurrence of events), then

Eq. 2.24 gives the joint PDF of the MPP defined by the Jacod likelihood which has no simultaneous events [13, 33, 24]. The case $M = 2$ corresponds to the joint PDF of a univariate point process [43].

MkPP likelihood

We show a new form of the continuous likelihood of the disjoint process above (Eq. 2.24) in terms of the MkPP representation. There are various ways we can arrive at this new form. We could start with the discrete-time likelihood expressed in terms of the discrete form of the MkPP representation A.13, divide by Δ^L, and let $\Delta \to 0$. This would amount to obtaining a *continuous* likelihood *from* an approximate *discrete* one by a limiting process similar to the previous derivation. Instead, we choose to start with the continuous likelihood of Eq. 2.24 and *re-arrange it in continuous-time* to obtain the continuous likelihood in terms of the MkPP representation:

$$p\big[N^*_{(0,T]}\big] = \prod_{m=1}^{M-1} \exp\left\{\int_0^T \log \lambda^*_m(t|H_t) dN^*_m(t) - \int_0^T \lambda^*_m(t|H_t)dt\right\} \tag{2.25}$$

$$= \prod_{m=1}^{M-1} \exp\left\{\sum_{\ell=1}^{L} \log \lambda^*_m(t_\ell|H_{t_\ell}) dN^*_m(t_\ell)\right\} \cdot \prod_{m=1}^{M-1} \exp\left\{-\int_0^T \lambda^*_m(t|H_t)dt\right\} \tag{2.26}$$

$$= \prod_{m=1}^{M-1}\prod_{\ell=1}^{L} \lambda^*_m(t_\ell|H_{t_\ell})^{dN^*_m(t_\ell)} \cdot \exp\left\{-\int_0^T \sum_{m=1}^{M-1} \lambda^*_m(t|H_t)dt\right\} \tag{2.27}$$

$$= \prod_{\ell=1}^{L}\prod_{m=1}^{M-1} \lambda^*_m(t_\ell|H_{t_\ell})^{dN^*_m(t_\ell)} \cdot \exp\left\{-\int_0^T \lambda^*_g(t|H_t)dt\right\} \tag{2.28}$$

$$= \prod_{\ell=1}^{L} \left(\frac{\lambda^*_g(t_\ell|H_{t_\ell})}{\lambda^*_g(t_\ell|H_{t_\ell})}\right)^{dN_g(t_\ell)} \prod_{m=1}^{M-1} \lambda^*_m(t_\ell|H_{t_\ell})^{dN^*_m(t_\ell)} \cdot \exp\left\{-\int_0^T \lambda^*_g(t|H_t)dt\right\} \tag{2.29}$$

$$= \prod_{\ell=1}^{L}\prod_{m=1}^{M-1} \left(\frac{\lambda^*_m(t_\ell|H_{t_\ell})}{\lambda^*_g(t_\ell|H_{t_\ell})}\right)^{dN^*_m(t_\ell)} \cdot \lambda^*_g(t_\ell|H_{t_\ell})^{dN_g(t_\ell)} \exp\left\{-\int_0^T \lambda^*_g(t|H_t)dt\right\}. \tag{2.30}$$

Chapter 3

Rescaling SEMPPs

In the preceding chapter, we showed that the continuous-time likelihood of $N^*(t)$ factorizes into the product of uni-variate point process likelihoods. In this chapter, after recalling the time-rescaling result for uni-variate point processes, we state results on rescaling multivariate point processes (with no simultaneities) [29, 11, 14, 46] to $N^*(t)$. The main implication of these results is that $N^*(t)$ can be mapped to a multivariate point process with independent unit-rate Poisson processes as its components. We apply the multivariate time-rescaling theorem to goodness-of-fit assessment for SEMPPs and describe several algorithms for simulating SEMPP models.

3.1 Rescaling uni-variate point processes

Time-Rescaling Theorem: *Let the strictly-increasing sequence* $\{t_\ell\}_{\ell=1}^L < T$ *be a realization from a point process* $N(t)$ *with conditional intensity function* $\lambda(t|H_t)$ *satisfying* $0 < \lambda(t|H_t)$ *for all* $t \in [0,T)$. *Define the transformation:*

$$\{t_\ell\} \rightarrow \{\Lambda(t_\ell)\} = \int_0^{t_\ell} \lambda(\tau|H_\tau)\mathrm{d}\tau\},$$

for $\ell \in \{1,\cdots,L\}$, *and assume* $\Lambda(t) < \infty$ *for all* $t \in [0,T)$. *Then the sequence* $\{\Lambda(t_\ell)\}_{\ell=1}^L$ *is a realization from a Poisson process with unit rate.*

According to the theorem, the sequence consisting of $\tau_1 = \Lambda(t_1)$ and $\{\tau_\ell = \Lambda(t_\ell) - \Lambda(t_{\ell-1})\}_{\ell=2}^L$ is a sequence of independent exponential random variables with mean 1. This is equivalent to saying that the sequence $\{u_\ell = 1 - \exp(-\tau_\ell)\}_{\ell=1}^L$ is a sequence

of independent uniform random variables on the interval $(0, 1)$ [9]. This first set of transformations allows us to check departure from the Poisson assertion of the theorem. If we further transform the u_ℓ's into $z_\ell = \Phi^{-1}(u_\ell)$ (where $\Phi(\cdot)$ is the distribution function of a zero mean Gaussian random variable with unit variance), then the theorem also implies that the random variables $\{z_\ell\}_{\ell=1}^{L}$ are mutually independent zero mean Gaussian random variables with unit variance. The benefit of this latter transformation is that it allows us to check independence by computing auto-correlation functions (ACFs). Next, we describe a procedure to assess the level of agreement between a fitted model, with estimated conditional intensity function $\hat{\lambda}(t|H_t)$, and the data.

Kolmogorov-Smirnov Test: The Kolmogorov-Smirnov test is a statistical test to assess the deviation of an empirical distribution from a hypothesized one. The test is implemented using a set of confidence bounds which depend on a desired confidence level (e.g. 95%, 99%), the sample size L and the hypothesized distribution (e.g. normal, uniform etc...). The test prescribes that the null hypothesis should be accepted if the empirical distribution lies within the confidence bounds specified by the theoretical model. The null hypothesis is the hypothesis that, with the desired confidence level, there is agreement between the data and the fit.

Recall that, according to the time-rescaling theorem, if the fitted model with conditional intensity function $\hat{\lambda}(t|H_t)$ fits the data then the sequence $\{\hat{u}_\ell\}_{\ell=1}^{L}$ is a sequence of independent uniform random variables on the interval $(0, 1)$. One can use the following KS GOF test to determine if the $\hat{u}_\ell$'s are indeed independent samples from a uniform random variable on the interval $(0, 1)$:

1. Order the $\hat{u}_\ell$'s from smallest to largest, to obtain a sequence $\{\hat{u}_{(\ell)}\}_{\ell=1}^{L}$ of ordered values.

2. Plot the values of the cumulative distribution function of the uniform density defined as $\{b_\ell = \frac{l-1/2}{L}\}_{\ell=1}^{L}$ against the $\hat{u}_{(\ell)}$'s.

If the model is correct, then the points should lie on the 45-degree line [23]. Confidence bounds can be constructed using the distribution of the KS statistic. For large enough L, the 95% and 99% confidence bounds are given by $b_\ell \pm \frac{1.36}{\sqrt{L}}$ and $b_\ell \pm \frac{1.63}{\sqrt{L}}$, respectively [23].

Testing for Independence of Rescaled times: One can assess the independence of the rescaled times by plotting the ACF of the $\hat{z}_\ell$ with its associated approximate confidence intervals calculated as $\pm\frac{z_{1-(\alpha/2)}}{\sqrt{L}}$ [5], where $z_{1-(\alpha/2)}$ is the $1-(\alpha/2)$ quantile of a Gaussian distribution with mean zero and unit variance.

An alternate application of the time-rescaling theorem is simulation of a univariate point processes [9]. This algorithm is a special case of one of the algorithms we describe in this chapter (Algorithm 2, with $M = 2$).

3.2 Rescaling multivariate point processes

We now state the time-rescaling result for "multivariate point processes" (Proposition 7.4.VI in [14]).

Proposition: *Let $N^*(t) = \{N^*_m(t) : m = 1, \cdots, M-1\}$ be a multivariate point process defined on $[0, \infty)$ with a finite set of components, full internal history H_t, and left-continuous H_t-intensities $\lambda^*_m(t|H_t)$. Suppose that for $m \in \{1, \cdots, M-1\}$ the conditional intensities are strictly positive and that $\Lambda^*_m(t) = \int_0^t \lambda^*_m(\tau|H_\tau)d\tau \rightarrow \infty$ as $t \rightarrow \infty$. Then under the simultaneous random time transformations:*

$$t \rightarrow \Lambda^*_m(t),\ m \in \{1, \cdots, M-1\},$$

the process $\{\left(N^*_1(t), \cdots, N^*_{M-1}(t)\right) : t \geq 0\}$ is transformed into a multivariate Poisson process with independent components each having unit rate.

Note: In the terminology of Vere-Jones et al., a "multivariate point process" refers to a vector-valued point process with *no* simultaneities. In this terminology, $N^*(t)$ would be considered a "multivariate point process" (by construction) while $N(t)$, as we have defined it in the previous chapter, in general would not. According to the

proposition, $N^*(t)$ can be transformed into a multivariate point process whose $M-1$ components are independent Poisson processes each having unit rate.

The proposition is a consequence of (a) the fact that the likelihood of $N^*(t)$ is the product of univariate point-process likelihoods, and (b) the time-rescaling result for uni-variate point processes. The interested reader should consult [14] for a rigorous proof.

Next, we discuss applications of the time-rescaling result of this section to simulation of SEMPPs and goodness-of-fit assessment respectively.

3.3 Application to simulation of SEMPPs

We present two classes of algorithms for simulating SEMPP models. The first class of algorithms uses the time-rescaling theorem (univariate or multivariate), while the second class uses thinning.

3.3.1 Algorithms based on the time-rescaling theorem

The following algorithm is based on the interpretation of SEMPPs as MkPPs with finite mark space: first we simulate from the ground process, then every time an event occurs, we roll an $M-1$-sided die.

Algorithm 1 (Time-rescaling): Given an interval $(0,T]$

1. Set $t_0 = 0$ and $\ell = 1$.
2. Draw u_ℓ from the uniform distribution on (0,1).
3. Find t_ℓ as the solution to: $\log(u_\ell) = \int_{t_{\ell-1}}^{t_\ell} \lambda_g^*(t|H_t)dt$.
4. If $t_\ell > T$, then stop, else
5. Draw m_ℓ from the (M-1)-dimensional multinomial distribution with probabilities $\frac{\lambda_m^*(t_\ell|H_{t_\ell})}{\lambda_g^*(t_\ell|H_{t_\ell})}$, $m = \{1, \cdots, M-1\}$.
6. set $dN_{m_\ell}^*(t_\ell) = 1$ and $dN_m^*(t_\ell) = 0$ for all $m \neq m_\ell$.

7. $dN(t_\ell)$ is obtained from $dN^*(t_\ell)$ using the map described in Chapter 2.

8. $\ell = \ell + 1$.

9. Go back to 2.

Note that step 3 of the above algorithm could be replaced by the following two steps: For each m, solve for t_ℓ^m as the solution to

$$\tau_\ell = \int_{t_{\ell-1}}^{t_\ell^m} \lambda_m^*(t|H_t)dt.$$

Then

$$t_\ell = \min_{m\in\{1,\cdots,M-1\}} t_\ell^m.$$

This follows from a known result which we derive below.

Suppose t_ℓ and $t_{\ell-1}$ are realization of some random variables T_ℓ and $T_{\ell-1}$ respectively, and that the t_ℓ^m's are realizations of random variables T_ℓ^m's, $m \in \{1, \cdots, M-1\}$:

$$\begin{aligned}
P[T_\ell \geq t_\ell | T_{\ell-1} = t_{\ell-1}] &= P[\min_m T_\ell^m \geq t_\ell | T_{\ell-1} = t_{\ell-1}] \\
&= \bigcap_{m=1}^{M-1} P[T_\ell^m \geq t_\ell | T_{\ell-1} = t_{\ell-1}] \\
&= \prod_{m=1}^{M-1} \exp\left\{\int_{t_{\ell-1}}^{t_\ell} \lambda_m^*(t|H_t)dt\right\} \\
&= \exp\left\{\sum_{m=1}^{M-1} \int_{t_{\ell-1}}^{t_\ell} \lambda_m^*(t|H_t)dt\right\} \\
&= \exp\left\{\int_{t_{\ell-1}}^{t_\ell} \left(\sum_{m=1}^{M-1} \lambda_m^*(t|H_t)\right) dt\right\} \\
&= \exp\left\{\int_{t_{\ell-1}}^{t_\ell} \lambda_g^*(t|H_t)dt\right\}.
\end{aligned}$$

The following algorithm for simulating SEMPPs follows from the time-rescaling result for $N^*(t)$. If there were no dependence of the CIFs on history, we would simulate observations from each component separately. However, due to history dependence, each component must inform other components to update their history as events

occur. Therefore, this algorithm is not as practical as the previous one. However, it follows directly from the the multivariate time-rescaling theorem discussed above.

Algorithm 2 (Time-rescaling):

1. Set $t_0 = 0$, $\ell = 1$, $\ell_m = 1 \; \forall \; m \in \{1, \cdots, M-1\}$.

2. $\forall \; m$, draw τ_{ℓ_m} an exponential random variable with mean 1.

3. $\forall \; m$, find t_{ℓ_m} as the solution to:
   $\tau_{\ell_m} = \int_{t_{\ell_m - 1}}^{t_{\ell_m}} \lambda_m^*(t|H_t)dt$.
   Let $m^+ = \arg\min_m t_{\ell_m}$, $t_\ell = t_{\ell_{m^+}}$.

4. If $t_\ell > T$, then stop the algorithm, else

5. If $m = m^+$, set $dN_{m^+}^*(t_\ell) = 1$, $\ell_m = \ell_m + 1$ and draw τ_{ℓ_m} an exponential random variable with mean 1.

6. If $m \neq m^+$, ℓ_m does not change, set
   $\tau_{\ell_m} = \tau_{\ell_m} - \int_{t_{\ell_m - 1}}^{t_\ell} \lambda_m^*(t|H_t)dt$,
   $t_{\ell_m - 1} = t_\ell$,
   $dN_m^*(t_\ell) = 0$,

7. $dN(t_\ell)$ is obtained from $dN^*(t_\ell)$ using the map described in Chapter 2.

8. $\ell = \ell + 1$.

9. Go back to 3.

3.3.2 Thinning-based algorithms

The following algorithm for simulating an SEMPP model is an extension of the thinning simulation algorithm for MPP models developed by Ogata [32].

Algorithm 3 (Thinning): Suppose there exists λ such that $\lambda_g^*(t|H_t) \leq \lambda$ for all $t \in (0, T]$:

1. Simulate observations $0 < t_1 < t_2 \cdots < t_K \leq T$ from a Poisson point process with rate λ.

2. Set $k = 1$.

3. while $k \leq K$

   (a) Draw u_k from the uniform distribution on (0,1)

   (b) if $\frac{\lambda_g^*(t_k|H_{t_k})}{\lambda} \leq u_k$

      i. Draw m_k from the $(M-1)$-dimensional multinomial distribution with probabilities $\frac{\lambda_m^*(t_k|H_{t_k})}{\lambda_g^*(t_k|H_{t_k})}$, $m = \{1, \cdots, M-1\}$

      ii. set $dN_{m_k}^*(t_k) = 1$ and $dN_m^*(t_k) = 0$ for all $m \neq m_k$

   (c) else, set $dN_m^*(t_k) = 0$ for all $m \in \{1, \cdots, M-1\}$

   (d) $dN(t_k)$ is obtained from $dN^*(t_k)$ as in Chapter 2.

   (e) $k = k + 1$.

An alternative form of Algorithm 3 is as follows:

Algorithm 4 (Thinning): Suppose there exists λ such that $\sum_{m=1}^{M-1} \lambda_m^*(t|H_t) \leq \lambda$ for all $t \in (0, T]$:

1. Simulate observations $0 < t_1 < t_2 \cdots < t_K \leq T$ from a Poisson point process with rate λ.

2. Set $k = 1$.

3. while $k \leq K$

   (a) Draw $m_k \in \{0, \cdots, M-1\}$ from the M-dimensional multinomial distribution with probabilities $\pi_0 = \frac{\lambda - \sum_{m=1}^{M-1} \lambda_m^*(t_k|H_{t_k})}{\lambda}$ and $\pi_m = \frac{\lambda_m^*(t_k|H_{t_k})}{\lambda}$, $m = 1, \cdots, M-1$

   (b) if $m_k = 0$, set $dN_m^*(t_k) = 0$ for all $m \in \{1, \cdots, M-1\}$

   (c) else, set $dN_{m_k}^*(t_k) = 1$ and $dN_m^*(t_k) = 0$ for all $m \neq m_k$

   (d) $dN(t_k)$ is obtained from $dN^*(t_k)$ as in Chapter 2.

   (e) $k = k + 1$.

Algorithms 3 and 4 are variations on the same algorithm. The former uses the fact that one can represent an M-nomial pmf as the product of a Bernoulli component and an $M-1$-nomial component.

3.3.3 Simulated joint neural spiking activity

We use the time-rescaling algorithm (Algorithm 1) to simulate simultaneous spiking activity from three thalamic neurons in response to periodic whisker deflections of velocity 50 mm/s. We simulate 33 trials of the experiment described in Chapter 5 using the following form for the CIFs:

$$\log\frac{\lambda_m^*[i|H_i]\Delta}{1-\lambda_g^*[i|H_i]\Delta} = \beta_{m,0} + \underbrace{\sum_{j=0}^{J-1}\beta_{m,j}^{(0)}s_{i-j}}_{\text{stimulus component}} + \underbrace{\sum_{c=1}^{3}\sum_{k=1}^{K_c}\beta_{m,k}^{(c)}\Delta N_{c,i-k}}_{\text{history component}}, \tag{3.1}$$

$m = 1, \cdots, 7$. In the next chapter, we will see that this parametric form of the CIFs gives a multinomial generalized linear model (mGLM). For these simulations, we chose $J = 2$, $K_1 = 2$, $K_2 = 2$ and $K_3 = 2$. We chose the parameters of the model based on our analysis, in Chapter 5, of the joint spiking activity of pairs of thalamic neurons in response to periodic whisker deflections of the same velocity.

Fig.3-1 shows the standard raster plots of the simulated data. There is strong modulation of the activity of each of the neurons by the stimulus. Fig. 3-2 shows the raster plots of each of the 7 disjoint components of ΔN^*. As the figure indicates, the parameters of the model were chosen so that the stimulus strongly modulates simultaneous occurrences from the pairs Neuron 1 and Neuron 2, Neuron 2 and Neuron 3, as well as simultaneous occurrences from the triple.

3.4 Application to goodness-of-fit assessment

Let $\{\Lambda_m^*(t_\ell)\}_{\ell=1}^{L_m}$ be the sequence obtained by rescaling points of $N^*(t)$ as in the multivariate time-rescaling theorem. There are L_m such points and the L_m's satisfy $\sum_{m=1}^{M-1} L_m = L$, where L is the total number of events from the ground process $N_g(t)$

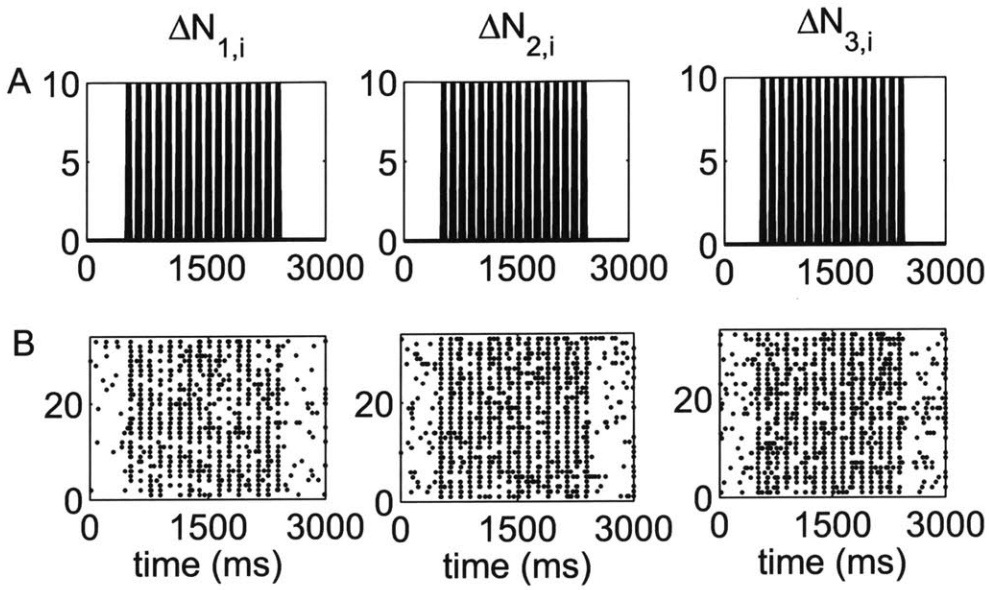

Figure 3-1. Standard raster plots of the simulated spiking activity of each neuron in a triplet in response to a periodic whisker deflection of velocity $v = 50$ mm/s. (A) Stimulus: periodic whisker deflection, (B) 33 trials of simulated data. The standard raster plots show that the stimulus induces strong modulation of the neural spiking of each of the three neurons. These standard raster plots do not clearly show the effect of the stimulus on joint spiking. The effect on the stimulus on joint spiking activity is evident in the new raster plots of the disjoint events (Fig. 3-2).

in the interval $[0, T)$. Now consider the sequence consisting of $\{\tau_1^m = \Lambda_m^*(t_1)\}$ and $\{\tau_\ell^m = \Lambda_m^*(t_\ell) - \Lambda_m^*(t_{\ell-1})\}_{\ell=2}^{L_m}$, $m \in \{1, \cdots, M-1\}$. According to the multivariate time-rescaling theorem, the τ_ℓ^m's ($\ell \in \{1, \cdots, L_m\}$, $m \in \{1, \cdots, M-1\}$) are mutually independent exponential random variables with mean 1. This is equivalent to saying that the random variables $\{u_\ell^m = 1-\exp(-\tau_\ell^m)\}_{\ell=1}^{L_m}$, $m \in \{1, \cdots, M-1\}$, are mutually independent uniform random variables on the interval $(0, 1)$. This latter fact forms the basis of a KS test for GOF assessment much like in the case of a uni-variate point process [9].

Kolmogorov-Smirnov Test: Assume that CIFs $\hat{\lambda}_m^*(t|H_t)$ were obtained by fitting a model to available data. **For each** m, one can use the following KS GOF test to determine whether or not the $\hat{u}_\ell^m$'s are samples from a uniform random variable on the interval $(0, 1)$:

1. Order the $\hat{u}_\ell^m$'s from smallest to largest, to obtain a sequence $\{\hat{u}_{(\ell)}^m\}_{\ell=1}^{L_m}$ of ordered values.

2. Plot the values of the cumulative distribution function of the uniform density defined as $\{b_\ell^m = \frac{l-1/2}{L_m}\}_{\ell=1}^{L_m}$ against the $\hat{u}_{(\ell)}^m$'s.

   If the model is correct then, for each $m \in \{1, \cdots, M-1\}$, the points should lie on the 45-degree line [23]. Confidence bounds can be constructed using the distribution of the KS statistic. For large enough L_m, the 95% and 99% confidence bounds are given by $b_\ell^m \pm \frac{1.36}{\sqrt{L_m}}$ and $b_\ell^m \pm \frac{1.63}{\sqrt{L_m}}$, respectively [23].

Testing for Independence of Rescaled Times :

If we further transform the u_ℓ^m's into $z_\ell^m = \Phi^{-1}(u_\ell^m)$ (where $\Phi(\cdot)$ is the distribution function of a zero mean Gaussian random variable with unit variance), then the proposition asserts that the random variables $\{z_\ell^m\}_{\ell=1}^{L_m}$ are mutually independent zero mean Gaussian random variables with unit variance. That is (a) for fixed m, the elements of $\{z_\ell^m\}_{\ell=1}^{L_m}$ are i.i.d. zero mean Gaussian with unit variance, (b) $\{z_\ell^m\}_{\ell=1}^{L_m}$ and $\{z_\ell^{m'}\}_{\ell=1}^{L_{m'}}$ are independent sets of random variables, $m \neq m'$. The benefit of this transformation is that it allows us to check independence by computing auto-correlation functions (ACFs) (for fixed m) and cross-correlation functions (CCFs) ($m \neq m'$).

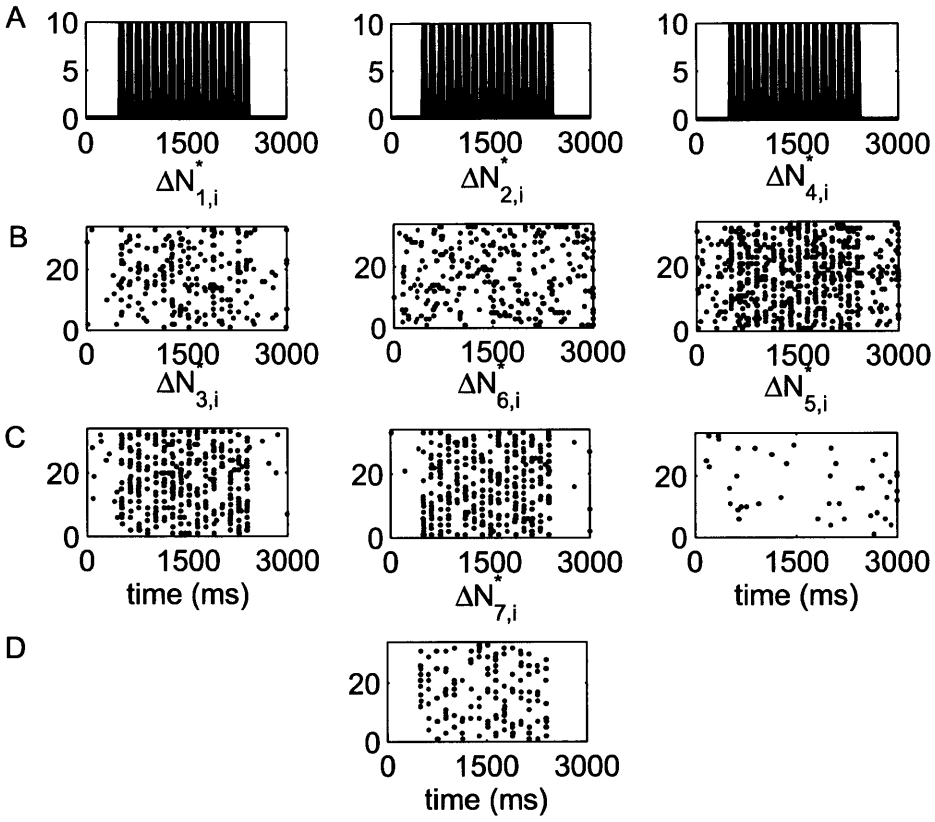

Figure 3-2. New raster plots of non-simultaneous ('100', '010' and '001') and simultaneous ('110', '011', '101' and '111') spiking events for the three simulated neurons of in Fig. 3-1. (A) Stimulus (B) Non-simultaneous events, from left to right, '100', '010' and '001', (C) Simultaneous events from pairs of neurons, from left to right, '110', '011' and '101', (D) Simultaneous event from the three neurons ('111'). The new raster plots of the three components show clearly the effects of the stimulus on non-simultaneous and simultaneous spiking. The $\Delta N^*_{4,i}$ and $\Delta N^*_{5,i}$ components of ΔN^* show that the joint spiking activity of the pairs consisting of Neurons 1 and 2 on the one hand, and Neurons 2 and 3 on the other hand is pronounced. The $\Delta N^*_{7,i}$ component of ΔN^* shows that the joint spiking activity of the three neurons is also pronounced. The information in these raster plots about the joint spiking activity of neurons could not be gathered from Fig. 3-1.

Chapter 4

Static and Dynamic Inference

In this chapter, we consider the problem of static and dynamic modeling of SEMPP data. For static inference, we propose a multinomial generalized linear model (mGLM) of the discrete-time likelihood of such data. For small enough sampling interval, the mGLM is equivalent to multiple Bernoulli GLMs. We perform estimation by maximizing the likelihood of the data using Newton's method. The use of linear conjugate gradient at each Newton step leads to fast algorithms for fitting the GLMs. For dynamic inference, we derive recursive linear filtering procedures to track a *hidden* parameter based on *observed* SEMPP data. In particular, we derive a multinomial adaptive filtering procedure, which uses the exact likelihood of the discrete-time representation of SEMPP. Using the approximate likelihood, we obtain generalizations of point-process adaptive filters.

4.1 Static modeling

We refer to *static* models as those for which the parameters of interest are fixed for a given set of observed data. For example, we classify the problem of fitting a line to data as a static modeling problem because we are seeking a *single* slope and intercept pair for the available data. However, we would not consider a static model one where we allow the slope and intercept to change (e.g. using an AR model).

We start with the likelihood of an SEMPP in discrete-time (Equation 2.12) and parametrize it so that it becomes a GLM with M-nomial observations and logit link. For small enough Δ, the mGLM is equivalent to $M-1$ separate uni-variate GLMs with Bernoulli observations and log link.

4.1.1 Generalized linear model of the DT likelihood

We may rewrite the discrete likelihood $P\left[\Delta N^*_{[1:I]}\right]$ of Equation 2.12 as follows:

$$\prod_{i=1}^{I} \exp\left\{\sum_{m=1}^{M-1} \Delta N^*_{m,i} \log \frac{\lambda^*_m[i|H_i]\Delta}{1-\lambda^*_g[i|H_i]\Delta} + \log(1-\lambda^*_g[i|H_i]\Delta)\right\}, \tag{4.1}$$

where we have substituted

$$\lambda^*_m[i|H_i] = P\left[\Delta N^*_{m,i} = 1 | \Delta N_{g,i} = 1, H_i\right] \lambda^*_g[i|H_i], m = 1, \cdots, M-1. \tag{4.2}$$

and dropped the $o(\Delta^L)$ component. The following relationships turn the above likelihood into a GLM with M-nomial observations and logit link [16]:

$$\log \frac{\lambda^*_m[i|H_i]\Delta}{1-\lambda^*_g[i|H_i]\Delta} = \beta'_m x_i, \text{ where} \tag{4.3}$$

β_m is a d-dimensional vector of parameters to be estimated from the data, x_i is a vector of covariates/features of the same dimension as β_m and $m = 1, \cdots, M-1$. The choice of covariates x_i depends on the problem at hand. In the case of neural data, the covariates are chosen so that they capture the effect of the stimulus as well as history on the observed response(s). The history portion comprises of past observations while choosing the stimulus depends on the experiment. It is easy to obtain an expression for $\lambda^*_m[i|H_i]\Delta$ as a function of $\beta_1, \cdots, \beta_m$ and x_i:

$$\lambda^*_m[i|H_i]\Delta = \frac{\exp\{\beta'_m x_i\}}{1+\sum_{m=1}^{M-1}\exp\{\beta'_m x_i\}}, m = 1, \cdots, M-1. \tag{4.4}$$

In the case of a bi-variate SEMPP $N(t)$ ($M = 2$), we may recover the marginal probabilities as

$$\lambda_1[i|H_i]\Delta = \lambda^*_1[i|H_i]\Delta + \lambda^*_3[i|H_i]\Delta, \tag{4.5}$$

$$\lambda_2[i|H_i]\Delta = \lambda^*_2[i|H_i]\Delta + \lambda^*_3[i|H_i]\Delta. \tag{4.6}$$

One may also recover the marginal probabilities in case of a C-dimensional SEMPP

$(M = 2^C)$ by using the map described in Chapter 2.

We are now in a position to write the parametrized joint likelihood of the DT process as a function of $\beta_1, \cdots, \beta_m$, the ΔN_i^*'s and the x_i's:

$$P\left[\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\right] = \exp\left\{\sum_{i=1}^{I}\sum_{m=1}^{M-1} \Delta N^*_{m,i}\beta'_m x_i - \log\left(1 + \sum_{m=1}^{M-1}\exp\{\beta'_m x_i\}\right)\right\}, \tag{4.7}$$

where $\beta = (\beta'_1, \cdots, \beta'_{M-1})'$.

The corresponding log-likelihood $L\left(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\right)$ is given by:

$$L\left(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\right) = \sum_{i=1}^{I}\sum_{m=1}^{M-1} \Delta N^*_{m,i}\beta'_m x_i - \log\left(1 + \sum_{m=1}^{M-1}\exp\{\beta'_m x_i\}\right). \tag{4.8}$$

Approximate GLM: For small Δ, the discrete-time likelihood is approximately the product of $M-1$ discrete-time univariate point-process likelihoods (Equation 2.18). Assuming $\sum_{m=1}^{M-1}\lambda^*_m[i|H_i]\Delta \propto o(1)$, we may write:

$$\log \lambda^*_m[i|H_i]\Delta \approx \log\frac{\lambda^*_m[i|H_i]\Delta}{1 - \sum_{m=1}^{M-1}\lambda^*_m[i|H_i]\Delta}. \tag{4.9}$$

If we let $\log \lambda^*_m[i|H_i]\Delta = \beta'_m x_i$ (in the approximate discrete likelihood of Equation 2.18), then the multinomial GLM is approximately equivalent to $M-1$ univariate GLMs with Bernoulli observations and log link. The corresponding likelihood and log-likelihood are given by:

$$P\left[\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\right] \approx \exp\left\{\sum_{i=1}^{I}\sum_{m=1}^{M-1} \Delta N^*_{m,i}\beta'_m x_i - \exp\{\beta'_m x_i\}\right\}, \tag{4.10}$$

$$L\left(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\right) \approx \sum_{i=1}^{I}\sum_{m=1}^{M-1} \Delta N^*_{m,i}\beta'_m x_i - \exp\{\beta'_m x_i\}. \tag{4.11}$$

4.1.2 Maximum likelihood estimation

Iteratively-reweighted least-squares (IRwLS)

Our objective function is the log likelihood of the data, given in Equation 4.8, which we would like to maximize. That is, we would like to find:

$$\hat{\beta}_{ML} = \underset{\beta}{\operatorname{argmax}}\ L\big(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\big). \tag{4.12}$$

This is a well-studied problem in the statistics and machine learning literatures, where it is known as logistic regression [27, 26, 4, 30].

In the appendix, we derive the gradient vector $g(\beta)$ and the Hessian matrix $H(\beta)$ of the objective as a function of $\beta = (\beta_1', \cdots, \beta_{M-1}')'$. It is not hard to show that, if the matrix of covariates/features is full-rank, then the Hessian matrix is negative definite. In turn, this implies that the ML estimate $\hat{\beta}_{ML}$ of β is unique.

We maximize the objective function by taking Newton steps as follows:

$$\begin{aligned} \beta_{(k+1)} &= \beta_{(k)} - H^{-1}(\beta_{(k)})g(\beta_{(k)}) && (4.13)\\ &= \beta_{(k)} + (X'W(\beta_{(k)})X)^{-1}X'\left(\Delta N^* - \lambda^*[\beta_{(k)}]\Delta\right). && (4.14) \end{aligned}$$

Various stopping criteria can be used. Typically, one stops after a given number of iterations or if the deviance has reached some threshold. The deviance is the log of the ratio of likelihoods between a saturated model and the one estimated from the data [16]. It generalizes the mean-squared error in the case of Gaussian observations (linear least-squares). In our case, the deviance is given by:

$$D\big(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\big) = -2L\big(\Delta N^*_{[1:I]}, x_{[1:I]}; \beta\big). \tag{4.15}$$

Minimizing the deviance is equivalent to maximizing the likelihood of the data.

Rearranging Equation 4.14 reveals some structure in each Newton step:

$$X'W(\beta_{(k)})X\beta_{(k+1)} = X'W(\beta_{(k)})X\beta_{(k)} + X'\left(\Delta N^* - \lambda^*[\beta_{(k)}]\Delta\right) \quad (4.16)$$

$$= X'W(\beta_{(k)})\left(X\beta_{(k)} + W^{-1}(\beta_{(k)})(\Delta N^* - \lambda^*[\beta_{(k)}]\Delta)\right) \quad (4.17)$$

As $X'W(\beta_{(k)})X$ is a symmetric positive-definite matrix, Equation 4.17 can be interpreted as a weighted least-squares (WLS) problem:

$$\beta_{(k+1)} = \underset{x}{\operatorname{argmax}} - \frac{1}{2}(b - Ax)'Q(b - Ax), \quad (4.18)$$

$b = X\beta_{(k)} + W^{-1}(\beta_{(k)})(\Delta N^* - \lambda^*[\beta_{(k)}]\Delta)$, $Q = W(\beta_{(k)})$ and $A = X$. The interpretation of each Newton step as in Equation 4.18 is the reason why the likelihood maximization algorithm described above is often referred to as IRwLS [16].

Linear Conjugate Gradient

Maximizing the log-likelihood of the data by IRwLS can be computationally very expensive [26]. That is why, the algorithm can be very slow at times, especially for large data sets [26], [30]. We saw in Equation 4.17 that each Newton step amounts to solving a linear system, which can also be interpreted as a (WLS) or quadratic optimization problem with negative definite Hessian (Equation 4.18). Treating each Newton step as a concave quadratic optimization problem allows one to consider use the linear conjugate gradient (CG) method [38]. The computational complexity of linear CG is proportional to the sparsity of the Hessian matrix [38]. As our experience and that of others ([26]) reveals, the use of CG often results in a significant boost in performance. Typically, only a small number of CG iterations are required at each Newton step in order to obtain an accurate enough solution [26]. In neuroscience applications in particular (where the covariates x_i include past observations), the covariate/feature matrix X is often very sparse, which in turn results in a sparse Hessian. The corresponding algorithm, where the linear system involved at each Newton step is only solved approximately, falls within the class of truncated Newton methods [26].

Linear CG is an iterative algorithm to solve $n \times n$ linear systems of the form $Ax = b$, where A is a symmetric positive definite matrix [38]. One can think of this linear system as having arisen from the quadratic program:

$$\min_x f(x) = \frac{1}{2}x'Ax - b'x + c. \tag{4.19}$$

Consider an iterative algorithm to solve the above quadratic program. Let $r_{(k)} = b - Ax_{(k)}$ be the gradient of $f(x)$ evaluated at $x_{(k)}$, where $x_{(k)}$ is an estimate of the minimizer of $f(x)$ at iteration k. One can interpret the linear CG algorithm as an iterative algorithm which generates search directions $d_{(k)}$ by A-conjugate Gram-Schmidt orthogonalization of the residuals $r_{(k)}$ [38]. The generated directions $d_{(k)}$ are A-conjugate, that is, they satisfy:

$$d'_{(i)}Ad_{(j)} = 0, \ i \neq j \in \{1, \cdots, n\}. \tag{4.20}$$

It can be shown that, by taking successive steps in the $d_{(k)}$ directions, the resulting algorithm known as linear CG would need at most n iterations to converge. This is because, unlike an algorithm such as steepest descent which may visit some of the $r_{(k)}$ directions multiple times, linear CG only visits each direction $d_{(k)}$ once [38].

While the interpretation of CG as A-conjugate Gram-Schmidt orthogonalization on the residuals $r_{(k)}$ is valid, it should not be taken too far. Strictly speaking, each step of the Gram-Schmidt process would require one to store all previously-generated directions, as these would be needed in the next step of the process (to compute the new search direction). This would make the algorithm very expensive both computationally and in terms of storage space required. Needless to say that, implemented in this fashion, linear CG would be unattractive. What makes linear CG attractive however is that, at any given iteration k, one is *only* required to *store* the *previous search direction* (instead of $k-1$ of them). Indeed, one can show that [38], at iteration k, $r_{(k)}$ is A-orthogonal to all previous search directions except $d_{(k-1)}$. This significantly reduces the computational and storage cost of the algorithm and is one of the reason why it has become so popular. A formal derivation of the previous result and other

properties of linear CG can be found in [38]. We now turn to how one would actually implement the CG algorithm in practice. The algorithm begins with an initial guess $x_{(0)}$ of the solution.

The linear CG algorithm:

$$d_{(0)} = r_{(0)} = b - Ax_{(0)}, \tag{4.21}$$

$$i = 0, \tag{4.22}$$

$$\text{Step 1}: \quad \alpha_{(i)} = \frac{r'_{(i)} r_{(i)}}{d'_{(i)} A d_{(i)}}, \tag{4.23}$$

$$x_{(i+1)} = x_{(i)} + \alpha_{(i)} d_{(i)}, \tag{4.24}$$

$$\text{Step 2}: \quad r_{(i+1)} = r_{(i)} - \alpha_{(i)} A d_{(i)}, \tag{4.25}$$

$$\text{Stop if desired accuracy reached, else} \tag{4.26}$$

$$\text{Step 3}: \quad \gamma_{(i+1)} = \frac{r'_{(i+1)} r_{(i+1)}}{r'_{(i)} r_{(i)}}, \tag{4.27}$$

$$d_{(i+1)} = r_{(i+1)} - \gamma_{(i+1)} d_{(i)} \tag{4.28}$$

$$i = i + 1, \tag{4.29}$$

$$\text{Step 4}: \quad \text{Go back to Step 1.} \tag{4.30}$$

The above algorithm, which utilizes only the previous search direction at each iteration, has $O(S)$ space and time complexity per iteration, where S is the number of non-zero entries of A. This is a significant improvement over the $O(n^2)$ (per-iteration) space and time complexity of an algorithm which would utilize all previous search directions at each iteration [38].

4.1.3 Numerical examples of savings due to linear CG

The data sets

We use several data sets from neuroscience experiments to demonstrate the computational savings that can be obtained by using linear CG at each Newton step. We compare our implementation of logistic regression with Matlab's native glmfit function.

Example 1 The data comes from neurons in the auditory system. The data was recorded from the auditory nerve of anesthetized cats following the presentation of the input sentence "Wood is best for making toys and blocks" spoken by a male and sampled at 10 kHz. The GLM for the data expresses the neuron's conditional intensity function as a function of the spectro-temporal properties of the input stimulus and neuron's history [35].

Example 2 The data is recorded from a neuron in awake macaque V1 while the animal was viewing a natural scenes movie. The same movie was presented multiple times. The GLM of the data uses basis splines to non-parametrically model the stimulus component of the neuron's conditional intensity function, similar to a PSTH. Further, spike history effects were included as a basis-spline-based autoregressive model [21].

Example 3 The data comes from one neuron in the rat thalamus and was recorded in response to a periodic whisker deflection of velocity 16 mm/s administered at 8 Hz for a period of 2000 ms. A delay period of 500 ms preceded and followed each trial. A total of 50 trials were recorded. The experiment was described in detail in [42]. In this example, we only use 33 of these trials. The GLM for the data relates the conditional intensity function of the neuron to the administered whisker stimulation and the neuron's firing history.

Example 4 The data comes from one neuron in a patient with Parkinson's disease while the patient performs a behavioral task. The patient is requested to move a joystick in one of 4 different directions (Up, Down, Right, Left). Recordings start 250 ms before movement onset and stop when the movement begins. The experiment and the model were described in detail in [37]. The GLM of the data models the neuron's conditional intensity function as a function of 4 categorical variables as well as the history of the neuron. Each categorical variables corresponds to one direction of the joystick.

Results

We call our implementation 'bnlrCG', which stands for binomial logistic regression with conjugate gradient. We have also implemented the Poisson GLM with conjugate gradient. We compare the algorithms to Matlab's glmfit based on running time in seconds, as well as the deviance of the model for several data sets from neuroscience experiments.

The examples where run on a machine with two dual core Intel processors at 2.83 and 3.01 GHz dual-core, 3GB of RAM, 32 bit Windows Vista and Matlab version R2008a.

The algorithm described in [26] uses a fixed number of CG iterations. However, when the number of covariates d is O(100), our experience with neural data shows that a number of CG iterations approximately equal to $\frac{d}{2}$ results in better fits, although not significantly so. In what follows, we refer to the size of problems we consider as $n \times d$, where n is the number of observations.

In interpreting the results of Table 4.1, one should note that 'bnlrCG' does not solve the GLM/logistic regression problem exactly: the algorithm leads to very good approximate solutions to the logistic regression problem at a fraction of the time required by solving the exact problem naively [26]. In the case of neural data, this is achieved while preserving goodness-of-fit as measured by KS plots [9]. These computational savings are important, considering that one must usually select among several competing models, which involves fitting of multiple GLMs.

Table 4.1 summarizes the results. In all but the 1st example, 'bnlrCG' and Matlab's glmfit function result in the same value of the deviance. The goodness-of-fits using either methods, as measured by KS plots [9], were indistinguishable. For the first example, the negligible discrepancy between the deviances is the result of truncating the Newton steps by using only a fraction of the number of required CG iterations required to solve the linear system exactly. However, the running time of 'bnlrCG' is much smaller than that of glmfit.

We do not report results for our implementation of the Poisson GLM because savings in running time are practically indistinguishable from the ones reported in

Table 4.1. Comparison of glmfit and bnlrCG on various neuroscience data sets

	Ex 1		Ex 2		Ex 3		Ex 4	
	$n = 60000$ $d = 500$		$n = 100000$ $d = 128$		$n = 88000$ $d = 43$		$n \approx 19000$ $d = 28$	
	Dev	Time (s)	Dev	Time (s)	Dev	Time (s)	Dev	Time (s)
glmfit	12425	230	51187	89	14416	45	7847.5	5.2
bnlrCG	12678	38	51187	10	14416	15	7847.5	0.8

Table 4.1.

The speed up due to linear CG is better appreciated if one considers situation where the need to fit a large number of GLMs arises. Typically, selecting a GLM of a given data set requires one to compare several models of the data. In the process, one would need to fit several GLMs of the data (e.g. up to 1000). The number of such GLM fits depends on the size of the parameter space over which the set of competing models lie. In the case of Ex 2., bnlrCG could reduce the time required to select the best model from 24 hours to 3hours. The need to fit several GLMs also arises in instances when one is interested in computing bootstrap confidence-interval estimates.

The implementation based on CG has made fitting of a large number of GLMs computationally tractable in cases when it previously wasn't using Matlab's native routine.

4.2 Dynamic modeling: SEMPP adaptive filters

A model is called *dynamic* if it is not static. In other words, we allow the parameters of interest to evolve in a constrained fashion. For example, consider the problem of fitting a 'line' to data. We would consider a *dynamic* model one where we allow the slope and intercept pair of the line to evolve (e.g. according to an AR model).

Adaptive filtering is a branch of signal processing which deals with tracking of a time-varying latent signal based on a given set of observations or measurements. The Kalman filter is a powerful adaptive filtering algorithm which, under certain as-

sumptions (Gauss-Markov process), is able to reliably track an underlying continuous-valued signal based on continuous-valued observations. The Kalman filter, however, is not very useful when the observations are binary valued as in the case of point process observations. This has prompted the design of adaptive filters tailored to point process observations, known as point process adaptive filters [15]. Point process adaptive filters have been shown to be very useful at various decoding (tracking tasks) in the context of neural data [10]. Conventional point-process adaptive filters are inherently unable to handle multivariate point process data with simultaneities. This is because they either assume a no-simultaneity, Jacod likelihood model of the data or independence. As previously argued, both of these assumptions are at best theoretically-convenient.

Here, we introduce SEMPP adaptive filters as natural generalizations of point process adaptive filters. Unlike conventional point-process adaptive filters [15], SEMPP adaptive filters are able to exploit simultaneous occurrences of events. We use the Bayes' rule Chapman-Kolmogorov framework along with a linear state equation and SEMPP observation models to derive adaptive filters appropriate for estimation from multivariate point processes with simultaneities. The adaptive filters which we derive closely resemble those introduced by [15]. In fact, the steps involved in the derivation are exactly the same. However, the key difference is the fact that the authors in [15] do not allow for co-occurrences in the original C-variate point process $N(t)$. The disjoint representation $N^*(t)$ of $N(t)$ allows us to account for co-occurrences and leads to simple, elegant filters. We only sketch the key steps of the derivation and refer the interested reader to the treatment in [15] for details.

Naturally, these new filters could be applied to decoding problems based on SEMPP observations. SEMPP adaptive filters could also be useful in the context of fitting models with time-varying parameters to SEMPP data. Estimation of models with time-varying parameters is typically performed using the EM algorithm or Monte-Carlo methods. SEMPP adaptive filters could be used in the E-step of an EM algorithm. Indeed, they would allow for approximate analytic computation of posterior density of the state (the parameters) given all observations up to a given

time.

Notation: We observe samples of a C-variate point process $N(t)$ in the interval $[0, T)$. Assuming that such a process possesses $M-1$ degrees of freedom at each $t \in [0, T)$, let $N^*(t)$ be its disjoint representation with $M-1$ components $C+1 \leq M \leq 2^C$. $N^*(t)$ is characterized by conditional intensity functions $\lambda^*_m(t|\theta_t, H_t)$ $(m \in \{0, \cdots, M-1\})$, which share are common *hidden* or *unknown* time-varying parameter vector θ_t. Note that, in this formulation, it is not hard to incorporate a *known* exogenous signal x_t on which the CIFs depend. However, we omit this to keep the notation simple. Switching to a discrete-time representation (with fine enough sampling interval Δ and a number I of observations) $N^*(t)$ is represented by indicator vectors ΔN^*_i, where $i = 1, \cdots, I$ is the discrete time index. The time-varying parameter θ_t is also discretized and represented by the vector θ_i. We let $\Delta N^*_{[1:i]} = [\Delta N^*_1, \cdots, \Delta N^*_i]$ denote the observations up to time i. To simplify notation, we aggregate all of the history that is pertinent to the probabilities of events at i into a common term $H_i = [\Delta N^*_{[1:i-1]}]$.

First, we derive adaptive filters based on the approximate DT likelihood for a single observation (Equation 2.18 with $I = 1$). These filters turn out to be generalizations of the ones introduced in [15]. Then, we derive filters based on the exact DT likelihood for a single observation(Equation 4.1 with $I = 1$) and argue that the former filters are approximations to the latter.

4.2.1 Adaptive filters based on approximate discrete-time likelihood

To develop an adaptive filter, we derive a recursive expression for θ_i in terms of its previous values and H_i. Time-varying estimates of θ_i will be based on its posterior density conditioned on past observations and H_i, $p(\theta_i|\Delta N^*_i, H_i)$. This posterior density evolves over time with each incoming observation. Tracking the evolution of this posterior density over time allows for tracking of the evolution of the parameter θ_i based on observations up to and including the current one at time i.

Before outlining the major steps involved in the derivation of the adaptive filters, we specify the system and observation equations. We define the system equation as

as a first-order vector auto-regressive process with Gaussian errors:

$$\theta_i = F_i \theta_{i-1} + \epsilon_i, \tag{4.31}$$

where F_i is a system evolution matrix and the ϵ_i's are i.i.d. zero-mean Gaussian random vectors with diagonal covariance matrices Σ_i for each i. This model imposes a stochastic continuity constraint on the θ_i's. Loosely, this model implicitly states that the θ_i's do not change much from one time step to the next.

The second component for the construction of a recursive filtering procedure is the likelihood or observation model specified in Equation 2.18. To keep the notation consistent with that of [15], we denote the likelihood by $p(\Delta N_i^* | \theta_i, H_i)$:

$$p(\Delta N_i^* | \theta_i, H_i) \approx \exp\left\{ \sum_{m=1}^{M-1} \Delta N_{m,i}^* \left(\log \lambda_m^*[i|\theta_i, H_i]\Delta \right) - \lambda_m^*[i|\theta_i, H_i]\Delta \right\}. \tag{4.32}$$

The recursive filtering equations which we seek are derived using the following procedure:

1. Bayes' rule to write in terms of likelihood and one-step prediction density:

$$p(\theta_i | \Delta N_i^*, H_i) = \frac{p(\Delta N_i^* | \theta_i, H_i) p(\theta_i | H_i)}{p(\Delta N_i^* | H_i)}. \tag{4.33}$$

   The first term of the numerator is the likelihood and the second term is the one-step prediction density defined by the Chapman-Kolmogorov equation:

$$p(\theta_i | H_i) = \int p(\theta_i, \theta_{i-1} | H_i) d\theta_{i-1} = \int p(\theta_i | \theta_{i-1}, H_i) p(\theta_{i-1} | H_i) d\theta_{i-1}. \tag{4.34}$$

   The above equation has two components: $p(\theta_i | \theta_{i-1}, H_i)$ given by the state evolution equation (Equation 4.31), and $p(\theta_{i-1} | H_i)$, the posterior density at the previous iteration.

2. Gaussian approximation to the posterior density: By assumption, $p(\theta_i | \theta_{i-1}, H_i)$ follows a Gaussian distribution in the parameter $\theta_i - \theta_{i-1}$. Approximating the

posterior density at the previous time step by a Gaussian implies that the one-step prediction density $p(\theta_i|H_i)$ also follows a Gaussian distribution. This is a simple consequence of the fact that the Chapman-Kolmogorov equation becomes the convolution of two Gaussians. Let $\theta_{i|i-1} = E[\theta_i|H_i]$ and $W_{i|i-1} = \text{var}[\theta_i|H_i]$ be the mean and covariance matrix of the one-step prediction density and $\theta_{i|i} = E[\theta_i|\Delta N_i^*, H_i]$ and $W_{i|i} = \text{var}[\theta_i|\Delta N_i^*, H_i]$ be the mean and variance of the posterior density. The Gaussian approximation of the posterior can then be expressed as follows:

$$\begin{aligned} p(\theta_i|\Delta N_i^*, H_i) &\propto \exp\left\{\sum_{m=1}^{M-1} \Delta N_{m,i}^* (\log \lambda_m^*[i|\theta_i, H_i]\Delta) - \lambda_m^*[i|\theta_i, H_i]\Delta\right\} \\ &\quad \cdot \exp\left\{-\frac{1}{2}(\theta_i - \theta_{i|i-1})' W_{i|i-1}^{-1}(\theta_i - \theta_{i|i-1})\right\} \qquad (4.35) \\ &\propto \exp\left\{-\frac{1}{2}(\theta_i - \theta_{i|i})' W_{i|i}^{-1}(\theta_i - \theta_{i|i})\right\}. \qquad (4.36) \end{aligned}$$

Taking the log on both sides yields:

$$\begin{aligned} -\frac{1}{2}(\theta_i - \theta_{i|i})' W_{i|i}^{-1}(\theta_i - \theta_{i|i}) &= \sum_{m=1}^{M-1} \Delta N_{m,i}^* (\log \lambda_m^*[i|\theta_i, H_i]\Delta) - \lambda_m^*[i|\theta_i, H_i]\Delta \\ &\quad - \frac{1}{2}(\theta_i - \theta_{i|i-1})' W_{i|i-1}^{-1}(\theta_i - \theta_{i|i-1}). \qquad (4.37) \end{aligned}$$

3. Solve for posterior mean and covariance: The recursive filtering equations are obtained by taking derivatives on both sides of the log equality above and evaluating at $\theta_i = \theta_{i|i-1}$. The interested reader is referred to [15] for the missing steps in the above outline of the derivation. Following the steps outlined above results in the recursive filtering equations:

$$\begin{aligned} \theta_{i|i-1} &= F_i \theta_{i-1|i-1}, \qquad (4.38) \\ W_{i|i-1} &= F_i W_{i-1|i-1} F_i' + \Sigma_i, \qquad (4.39) \end{aligned}$$

$$W_{i|i}^{-1} = W_{i|i-1}^{-1} + \sum_{m=1}^{M-1} \left[\left(\frac{\partial \log \lambda_{m,i}^*}{\partial \theta_i} \right)' [\lambda_{m,i}^* \Delta] \left(\frac{\partial \log \lambda_{m,i}^*}{\partial \theta_i} \right) \right]_{\theta_{i|i-1}} \quad (4.40)$$

$$- \sum_{m=1}^{M-1} \left[(\Delta N_{m,i}^* - \lambda_{m,i}^* \Delta) \frac{\partial^2 \log \lambda_{m,i}^*}{\partial \theta_i \partial \theta_i'} \right]_{\theta_{i|i-1}}, \quad (4.41)$$

$$\theta_{i|i} = \theta_{i|i-1} + W_{i|i} \sum_{m=1}^{M-1} \left[\left(\frac{\partial \log \lambda_{m,i}^*}{\partial \theta_i} \right)' (\Delta N_{m,i}^* - \lambda_{m,i}^* \Delta) \right]_{\theta_{i|i-1}}. \quad (4.42)$$

4.2.2 Adaptive filters based on exact discrete-time likelihood (multinomial filters)

The setup is the same as in the previous section, except that the approximate likelihood (Equation 2.18) for a single observation (i.e. $I = 1$) is replaced with the exact one (Equation 4.1):

$$\begin{aligned} p(\Delta N_i^* | \theta_i, H_i) &= \exp \left\{ \sum_{m=1}^{M-1} \Delta N_{m,i}^* \left(\log \frac{\lambda_m^*[i|\theta_i, H_i]\Delta}{1 - \lambda_g^*[i|\theta_i, H_i]\Delta} \right) + \log(1 - \lambda_g^*[i|\theta_i, H_i]\Delta) \right\} \\ &= \exp \left\{ \eta(\theta_i)' \Delta N_i^* - \log \left(1 + \sum_{m=1}^{M-1} \exp(\eta_m(\theta_i)) \right) \right\} \\ &= \exp \left\{ \eta(\theta_i)' \Delta N_i^* - A(\eta(\theta_i)) \right\}, \end{aligned} \quad (4.43)$$

where $\eta_m(\theta_i) = \log \frac{\lambda_m^*[i|\theta_i, H_i]\Delta}{1 - \lambda_g^*[i|\theta_i, H_i]\Delta}$.

Following the same reasoning as in the previous section, the Gaussian approximation to the posterior density of θ_i takes the following form:

$$\begin{aligned} p(\theta_i | \Delta N_i^*, H_i) &\propto \exp \left\{ \eta(\theta_i)' \Delta N_i^* - A(\eta(\theta_i)) \right\} \\ &\quad \cdot \exp \left\{ -\frac{1}{2} (\theta_i - \theta_{i|i-1})' W_{i|i-1}^{-1} (\theta_i - \theta_{i|i-1}) \right\} \quad (4.44) \\ &\propto \exp \left\{ -\frac{1}{2} (\theta_i - \theta_{i|i})' W_{i|i}^{-1} (\theta_i - \theta_{i|i}) \right\}. \quad (4.45) \end{aligned}$$

The recursive filtering equations are obtained by taking derivatives on both sides of the log equality above and evaluating at $\theta_i = \theta_{i|i-1}$. Taking the log on both sides

yields:

$$-\frac{1}{2}(\theta_i - \theta_{i|i})' W_{i|i}^{-1}(\theta_i - \theta_{i|i}) = \underbrace{\eta(\theta_i)'\Delta N_i^* - A\left(\eta(\theta_i)\right)}_{\text{exponential family term}} - \frac{1}{2}(\theta_i - \theta_{i|i-1})' W_{i|i-1}^{-1}(\theta_i - \theta_{i|i-1}). \tag{4.46}$$

The 'difficulty' in deriving the filtering equations comes from being able to differentiate the 1st term of the equation above with respect to θ_i. Fortunately, the observations ΔN_i^* belong to the exponential family of distributions, which possesses attractive properties. In particular, the following differential equalities are useful:

$$\begin{aligned}
\nabla_\eta A(\theta)|_{\eta(\theta_i)} &= E[\Delta N_i^*|\theta_i, H_i] = \lambda^*[i|\theta_i, H_i]\Delta \text{ and,} && (4.47)\\
\nabla_\eta^2 A(\theta)|_{\eta(\theta_i)} &= \text{Cov}(\Delta N_i^*, \Delta N_i^*|\theta_i, H_i) && (4.48)\\
&= \text{diag}\ \lambda^*[i|\theta_i, H_i]\Delta - \lambda^*[i|\theta_i, H_i]\lambda^*[i|\theta_i, H_i]'\Delta^2. && (4.49)
\end{aligned}$$

The above equalities imply that:

$$\begin{aligned}
\nabla_{\theta_i} \log p(\Delta N_i^*|\theta_i, H_i) &= \nabla'_{\theta_i}\eta(\theta_i)\left(\Delta N_i^* - \nabla_\eta A(\theta)|_{\eta(\theta_i)}\right)\\
&= \nabla'_{\theta_i}\eta(\theta_i)\left(\Delta N_i^* - \lambda^*[i|\theta_i, H_i]\Delta\right), \text{ and} && (4.50)\\
\nabla^2_{\theta_i} \log p(\Delta N_i^*|\theta_i, H_i) &= \sum_{m=1}^{M-1}\left(\Delta N_{m,i}^* - \lambda_m^*[i|\theta_i, H_i]\right)\nabla^2_{\theta_i}\eta_m(\theta_i) - \nabla'_{\theta_i}\eta(\theta_i)\cdot\nabla_\eta^2 A(\theta)|_{\eta(\theta_i)}\cdot\nabla_{\theta_i}\eta(\theta_i)\\
&= \sum_{m=1}^{M-1}\left(\Delta N_{m,i}^* - \lambda_m^*[i|\theta_i, H_i]\Delta\right)\nabla^2_{\theta_i}\eta_m(\theta_i)\\
&\quad -\nabla'_{\theta_i}\eta(\theta_i)\cdot\left(\text{diag}\ \lambda^*[i|\theta_i, H_i]\Delta - \lambda^*[i|\theta_i, H_i]\lambda^*[i|\theta_i, H_i]'\Delta^2\right)\cdot\nabla_{\theta_i}\eta(\theta_i), && (4.51)
\end{aligned}$$

where

$$\nabla_{\theta_i}\eta(\theta_i) = \begin{bmatrix} \nabla_{\theta_i}\eta_1(\theta_i)' \\ \cdot \\ \cdot \\ \nabla_{\theta_i}\eta_{M-1}(\theta_i)'. \end{bmatrix}$$

We are now in a position to write down the multinomial filtering equations:

$$\theta_{i|i-1} = F_i \theta_{i-1|i-1}, \tag{4.52}$$

$$W_{i|i-1} = F_i W_{i-1|i-1} F_i' + \Sigma_i, \tag{4.53}$$

$$W_{i|i}^{-1} = W_{i|i-1}^{-1} + \left[\nabla'_{\theta_i} \eta(\theta_i) \cdot \underbrace{\left(\text{diag } \lambda_i^* \Delta - \lambda_i^* \lambda_i^{*\prime} \Delta^2 \right)}_{\text{multinomial covariance}} \cdot \nabla_{\theta_i} \eta(\theta_i) \right]_{\theta_{i|i-1}} \tag{4.54}$$

$$- \sum_{m=1}^{M-1} \left[\left(\Delta N_{m,i}^* - \lambda_{m,i}^* \Delta \right) \nabla_{\theta_i}^2 \eta_m(\theta_i) \right]_{\theta_{i|i-1}}, \tag{4.55}$$

$$\theta_{i|i} = \theta_{i|i-1} + W_{i|i} \left[\nabla'_{\theta_i} \eta(\theta_i) \left(\Delta N_i^* - \lambda_i^* \Delta \right) \right]_{\theta_{i|i-1}}. \tag{4.56}$$

Equivalence with filters obtained using approximate likelihood: If (a) we let $\eta_m(\theta_i) \approx \log \lambda_m^*[i|\theta_i, H_i]\Delta$ and (b) assume that the off diagonal terms of the multinomial covariance matrix are $\propto o(\Delta)$, then we recover the filtering equations obtained in the previous subsection using the approximate likelihood.

Chapter 5

Data Analysis

In this chapter, we apply the machinery developed in the previous chapters to the analysis of simultaneous recordings from pairs of neurons in the rat thalamus, in response to repetitive whisker deflections of varying velocity. The recorded activity of theses pairs of neurons constitute a sample from of a bi-variate point process and hence are amenable to characterization using the techniques introduced in this thesis. Using these techniques, namely modeling of multivariate point processes in the GLM framework, we are able to provide an estimate of the extent to which whisker stimulation increases the propensity of pairs of thalamic neurons to fire simultaneously. We find that the effect of the stimulus on the simultaneous-spiking event can be in the same order as its effect on the non-simultaneous-spiking events. Surprisingly, for a number of the pairs, the former is even stronger than the latter. We also apply the dynamic-inference algorithms to decoding of whisker deflection velocity.

5.1 Thalamic firing synchrony in rodents

Rodents use rapid whisker movements to perform fine tactile discrimination. Thalamic neurons, which process tactile information from the whiskers, respond to single or periodic whisker deflections with a low mean firing rate [40, 20]. This had led neuroscientists to postulate that groups of thalamic neurons encode tactile information in the temporal proximity of the spikes which they emit, rather than single cell response magnitudes or interspike intervals. A population code based on firing synchrony would be well-suited for the task of detecting and processing rapid changes in whisker movements.

The case for the existence of a population code based on firing synchrony is supported by findings that cells in layer IV of visual and somatosensory cortex tend to respond to near-synchronous firing of their thalamic input neurons [2, 44, 36, 12]. In turn, these findings suggest that thalamic neurons play an important role in the selective transmission and processing of relevant sensory input.

Recent advances in our ability to record simultaneous spiking activity from multiple neurons [47, 28], have made it possible to directly investigate thalamic firing synchrony. In [42], the authors applied a cross-correlation analysis to simultaneous recordings from pairs of thalamic neurons in the same electrophysiologically-identified barreloid, in response to periodic whisker deflections of varying velocity. They report systematic changes in both onset time and strength of thalamic firing synchrony as a function of stimulus velocity.

Here, we use the likelihood-based point process approach developed in the previous chapters to investigate thalamic firing synchrony and its stimulus-dependent modulation. This approach offers several advantages over histogram-based methods such as cross-correlation analyses. First, whereas the results in [42] are obtained by averaging the responses of *all* pairs in the data set, we are able to characterize thalamic firing synchrony at the level of *individual* neuron pairs. Second, we are able to isolate the contribution of the stimulus, as opposed to neurons' intrinsic dynamics, to non-simultaneous and simultaneous (synchronous) events. We measure changes in the stimulus-induced modulation of thalamic firing synchrony as changes in the contribution of the stimulus to the instantaneous rate of the simultaneous-spiking event at the one ms time-scale. Last but not least, being likelihood-based, our inference framework carries all the optimality properties of the likelihood theory.

5.2 Experiment

We briefly describe the data set we analyze in this chapter. The experiments were previously described in detail in [42].

Simultaneous single-unit activity from pairs of thalamic neurons was recorded

with two electrodes placed in the same electrophysiologically-identified barreloid of the rat ventral posteromedial nucleus. Spiking activity was recorded from the pairs in response to whisker deflections at three different velocities administered at $8Hz$ for a period of 2000 ms. A delay period of 500 ms preceded and followed each stimulus period. The deflection velocities were 16, 50 and 80 mm/s. For each neuronal pair and each deflection velocity, the responses were recorded across 50 trials. We divided the 50 trials into a training set and a test set by randomly choosing 1 of every sequence of 3 trials and assigning it to the training set (17 trials). The remaining trials were assigned to the test set (33 trials).

Figs. 5-1A, 5-2A and 5-3A show standard raster plots of the data from a representative pair, respectively in response to stimulus velocities 16, 50 and 80 mm/s. These raster plots show that the stimulus (Figs. 5-1A, 5-2A and 5-3A, Row 1) induces strong modulation of the neural spiking in the training set (Figs. 5-1A, 5-2A and 5-3A, Row 2) and in the test set (Figs. 5-1A, 5-2A and 5-3A, Row 3), for *each* neuron in the pair. These figures, however, do not clearly show the *simultaneous* or *joint* spiking activity of the pair. To highlight the effect of the stimulus on the *joint* spiking activity of the pair, we introduce the new raster plots in Figs. 5-1B, 5-2B and 5-3B. Two of these raster plots (Figs. 5-1B, 5-2B and 5-3B, Columns 1 and 2) show the non-simultaneous activity of the pair, while the third (Figs. 5-1B, 5-2B and 5-3B, Columns 3) shows the simultaneous spiking activity of the pair. For simplicity, we refer to the non-simultaneous events as '01' and '10', reflecting the fact that they correspond to the cases where one of the neuron has an event *and* the other does not. Similarly, we refer to the simultaneous event as the '11' event, reflecting the fact it corresponds to the case where both neurons have an event. The new raster plots show that the stimulus induces a strong modulation of the *joint* spiking activity of the neurons in the pair. This was not apparent from the standard raster plots of Figs. 5-1A, 5-2A and 5-3A.

The goal of the analysis is to quantify the effect of the three stimuli on the joint spiking activity of each of the 17 pairs of neurons in the data set. The new raster plots of Figs. 5-1B, 5-2B and 5-3B provide a useful visual quantification of this effect.

The representation of the data in these new raster plots is also useful for data analysis purposes. Intuitively, this is because the new raster plots show *disjoint* events from the neuronal pair. Effectively, we have transformed a bivariate point process with simultaneous events (Figs. 5-1*A*, 5-2*A* and 5-3*A*) into a new trivariate point process of *disjoint* events (Figs. 5-1*B*, 5-2*B* and 5-3*B*) from the original bivariate process. The advantage of this transformation is that the new trivariate process is now amenable to standard point-process modeling techniques [13, 33].

5.3 Statistical model

We assume that the data constitute a sample from a bi-variate SEMPP, whose discrete-time likelihood can be written as a product of conditional four-nomial trials. As shown in Chapter 4, If we let

$$\log\frac{\lambda^*_m[i|H_i]\Delta}{1-\lambda^*_g[i|H_i]\Delta} = \beta_{m,0} + \underbrace{\sum_{j=0}^{J-1}\beta^{(0)}_{m,j}s_{i-j}}_{\text{stimulus component}} + \underbrace{\sum_{c=1}^{2}\sum_{k=1}^{K_c}\beta^{(c)}_{m,k}\Delta N_{c,i-k}}_{\text{history component}}, \qquad (5.1)$$

then the parametric model becomes a GLM with four-nomial observations and logit link. The model expresses the log odds of each outcome with respect to the base outcome as the convolution of the stimulus s with a finite length kernel $\{\beta^{(0)}_{m,j}\}_{j=0}^{J-1}$, and the history of ΔN_1 and ΔN_2 respectively with finite length kernels $\{\beta^{(1)}_{m,k}\}_{k=1}^{K_1}$ and $\{\beta^{(2)}_{m,k}\}_{k=1}^{K_2}$. Estimation is performed by maximizing the discrete-time likelihood of the data under the above parametric model, as detailed in Chapter 4. We select J, K_1 and K_2 using Akaike's information criterion:

$$AIC(J,K_1,K_2) = -2 * \log P\left[\Delta N^*_{[1:I]};\beta\right] + 2(J+K_1+K_2+1).$$

We assess GOF by time-rescaling as described in Chapter 3.

It should be noted that the stimulus s is the explicit or actual waveform that was administered during the experiment.

5.3.1 Measures of thalamic firing synchrony

For a *given pair* of neurons *and* an administered *stimulus*, we would like to extract meaningful information from the model of Eq. 5.1, which lends itself to interpretation. Below, we describe three quantities that arise from the model, which can be used to assess (a) the contribution of the stimulus to zero-lag synchrony, (b) overall zero-lag dependence between the neurons, and (c) overall non-zero lag dependence between the neurons. The latter is quantified in terms of the effects of the history of either neuron on the probability of the other neuron firing in the present.

Stimulus-induced modulation of thalamic firing synchrony

The advantage of our likelihood-based framework (Eq. 2.12) over existing histogram-based methods [8, 7, 17] is that it helps us to isolate the contribution of the stimulus, as opposed to neurons' history, to the joint events of the pair, i.e. '01', '10' and '11'. This model is also superior to existing point-process likelihood based methods [13, 33] because it allows us to characterize the joint spiking activity of the pair ('11' event) at any given recording resolution Δ.

For each joint event (i.e. $m = 1, 2, 3$), we define the stimulus-induced modulation of that event, that is the effect of the stimulus on that event, by:

$$SM_m[i] = \exp\{\sum_{j=0}^{J-1} \beta_{m,j}^{(0)} s_{i-j}\}. \tag{5.2}$$

This represents, in the i^{th} discrete-time bin, the amount by which the stimulus increases the instantaneous rate of each of the joint events at the Δ (one ms) time scale. The $m = 3$ component is of particular interest as it is the component of ΔN^* which represents joint spiking of the neurons in the pair. $SM_3[i]$ tells us how much the stimulus contributes to increasing the instantaneous rate at which the neurons in the pair fire simultaneously. We use this as a measure of stimulus-induced modulation of thalamic firing synchrony.

Equation 5.2 allows us to make a statement about the effect of the stimulus on zero-lag synchrony of the neurons. We now describe a measure of zero-lag synchrony that takes into account the effect of the dynamics of the neurons in the pair. We define

$$\rho[i] = \frac{\lambda_3^*[i|H_i]\Delta}{\sqrt{\lambda_1[i|H_i]\Delta(1-\lambda_1[i|H_i]\Delta)\lambda_2[i|H_i]\Delta(1-\lambda_2[i|H_i]\Delta)}}. \tag{5.3}$$

This quantity was used in [42] as a measure of firing synchrony: it reflects both the correlation caused by direct stimulus modulation of the two neurons' firing rates, as well as the correlation due to common input. Equation 5.3 is similar to the expression for the correlation coefficient of $\Delta N_{1,i}$ and $\Delta N_{2,i}$: the numerator is $E[\Delta N_{1,i}\Delta N_{2,i}|H_i]$ and the denominator is $\sigma_{\Delta N_{1,i}|H_i}\sigma_{\Delta N_{2,i}|H_i}$. It is not hard to show that the components of a bivariate Bernoulli random vector are independent if and only if they are uncorrelated. In each discrete-time bin, the model of Eq. 5.1 results in an estimate of a *joint* pmf, conditioned on history. Therefore, we can assess the time-varying dependence between the neurons in a pair using the (conditional) covariance in each time bin.

In [42], the authors compute the quantity of Eq. 5.3 at different lags, that is for different values of $j \neq 0$ and pairs $\Delta N_{1,i}$ and $\Delta N_{2,i+j}$. We do not compute these here as results using such estimates have already been reported in [42]. However, we explain below how the same quantities can be extracted from our model.

The CIFs $\lambda_m^*(t|H_t)$, $m = 1,2,3$ fully characterize the *joint* density of the vector process $(N_1(t), N_2(t))$. Equation 5.1 is a discrete-time model of this joint density. Once we fit the model and establish adequate goodness-of-fit, we can obtain (by marginalization of the joint density of ΔN^*) the joint probability mass function (PMF), for any time i and any lag j of any pair $(\Delta N_{1,i},\Delta N_{2,i+j})$. From the joint PMF, one could compute quantities similar to Eq. 5.3, now indexed by the lag j. Direct computation of the joint PMF of $\Delta N_{1,i}$ and $\Delta N_{2,i+j}$ is not tractable. Instead, it is more reasonable to simulate observations from the joint process using the estimated model parameters (Eq. 5.1) and the algorithms described in Chapter 3. The simulated data can then be used to compute quantities similar to the ones described

in [42]. The key point here is that any statistics of interests can be extracted from our model by virtue of the fact that we have an estimate of the joint density of the two neurons as a function of the stimulus.

We use the parameters of the model (Eq. 5.1) to assess the degree of non-zero lag dependence between the neurons. This is explained below.

Non-zero-lag dependence

In Eq. 5.1, $\beta_m^{(c)} = (\beta_{m,1}^{(c)}, \cdots, \beta_{m,K_c}^{(c)})'$ captures the effect of Neuron c on joint event m ($c = 1, 2$, $m = 1, 2, 3$). Recall that $m = 1$ corresponds to the '10' event, $m = 2$ to the '01' event and $m = 3$ to the simultaneous '11' event. Intuitively, a negative value of $\beta_{m,k}^{(c)}$ means that a spike in Neuron c that occurred a time $i - k$ ms will decrease the probability of event m at time i ms by $e^{\beta_{m,k}^{(c)}}$. Similarly, a positive value of $\beta_{m,k}^{(c)}$ means that a spike in Neuron c that occur-ed a time $i - k$ ms will increase the probability of event m at time i ms by $e^{\beta_{m,k}^{(c)}}$.

We characterize the effect of Neuron 1's history on its own probability of firing using a linear combination of $\beta_1^{(1)}$ and $\beta_3^{(1)}$. The effect of Neuron 2's history on the probability of Neuron 1 firing is obtained using a linear combination of $\beta_1^{(2)}$ and $\beta_3^{(2)}$. We characterize the effect of Neuron 2's history on its own probability of firing using a linear combination of $\beta_2^{(2)}$ and $\beta_3^{(2)}$. The effect of Neuron 1's history on the probability of Neuron 2 firing is obtained using a linear combination of $\beta_2^{(1)}$ and $\beta_3^{(1)}$.

Let n_{10} be the number of '10' events, n_{01} the number of '01' events and n_{11} the number of '11' events. Note that, because the events '10', '01' and '11' are disjoint, $n_1 = n_{10} + n_{11}$ and $n_2 = n_{01} + n_{11}$ represent the number of events respectively from Neuron 1 and Neuron 2. From $\beta_m^{(c)}$, we define the following quantities:

$$\begin{aligned}
\gamma_1^{(1)} &= \frac{n_{10}}{n_1}\beta_1^{(1)} + \frac{n_{11}}{n_1}\beta_3^{(1)} \\
\gamma_1^{(2)} &= \frac{n_{10}}{n_1}\beta_1^{(2)} + \frac{n_{11}}{n_1}\beta_3^{(2)} \\
\gamma_2^{(1)} &= \frac{n_{01}}{n_2}\beta_2^{(1)} + \frac{n_{11}}{n_2}\beta_3^{(1)} \\
\gamma_2^{(2)} &= \frac{n_{01}}{n_2}\beta_2^{(2)} + \frac{n_{11}}{n_2}\beta_3^{(2)},
\end{aligned}$$

where $\gamma_c^{(c')}$ now represents the effect of the history of Neuron c' on the probability of Neuron c firing in the present $(c, c' = 1, 2)$. This weighted linear combination of the coefficients makes intuitive sense because if $n_{11} = 0$, we obtain a characterization of the effect of the Neurons' history on their present which is the same as would be obtained from the Jacod-like approach (which assumes no simultaneous events) [33, 43].

These new coefficients can be interpreted as follows: a negative value of $\gamma_{c,k}^{(c')}$ means that a spike in Neuron c' that occurred a time $i - k$ ms will decrease the probability of Neuron c spiking at time i ms by $e^{\gamma_{c,k}^{(c')}}$. Similarly, a positive value of $\gamma_{c,k}^{(c')}$ means that a spike in Neuron c' that occurred a time $i - k$ ms will increase the probability of Neuron c spiking at time i ms by $e^{\gamma_{c,k}^{(c')}}$.

We used 17 trials of training data to fit the model of Eq. 5.1. The data suggests that the neurons' response to the stimulus does not vary across trials. That's why, our model is such that the parameters do not very across trials. This means that our characterization of the non-zero-lag dependence is the same across trials. However, the neuron's history changes from trial to trial. So, to compute $\rho[i]$, we first average the estimates of $\lambda_m^*[i|H_i]$ across the 17 trials. Since the stimulus is periodic and the same for all trials, we only need to compute $SM_m[i]$ for one stimulus cycle.

5.4 Results

We discuss in detail results for the pair displayed in Figs. 5-1, 5-2 and 5-3. We also show results for another representative pair in the data set and end the section with a summary of results for the entire data set analyzed.

5.4.1 Results for individual pairs

To select the optimal model order for each pair, we considered values for J, K_1 and K_2 ranging from 2 to 50 ms, in 1 ms increments. We used the results of preliminary GLM analyses on each neuron separately to reduce the dimension of the search space. We found that reducing J to a value as low as $J = 2$ did not affect the goodness-of-fit,

as measured by the number of points outside of the 95% confidence bounds in the KS plots. Therefore, for all pairs, the results we report here are for $J = 2$

The KS plots show that the model fits both the training (Figs. 5-4A, 5-5A, 5-6A) and test data (Figs. 5-4B, 5-5B, 5-6B) well, at all velocities. The good KS performance on each of the components of ΔN^* demonstrates the model's accurate description of the *joint* process. The performance on the test data demonstrates the strong predictive power of the model.

Fig. 5-7A compares the modulation of the non-simultaneous and simultaneous events by the stimulus for each of the three stimulus velocities. The figure shows that the stimulus modulates each of the simultaneous and non-simultaneous events at all velocities. Moreover, for the high and medium-velocity stimuli, the stimulus modulation of the '11' event is on the same order as that of the '01' event and much stronger than that of the '10' event. However, for the low-velocity stimulus, the modulation of the non-simultaneous events is stronger than that of the simultaneous event. In short, the stimulus induces zero-lag thalamic firing synchrony for all three stimuli. As is clearer from Fig. 5-8A, zero-lag stimulus-induced thalamic firing synchrony, as measured by the stimulus modulation of the '11' event, is much stronger for the high and medium-velocity stimuli. Indeed, the figure (which compares the stimulus modulation of the '11' event across stimuli), suggests that the stimulus modulation of the '11' event by the high and medium-velocity stimuli is two orders of magnitude stronger than the modulation by the low-velocity stimulus. The higher the velocity, the stronger the effect of the stimulus on simultaneous firing.

Fig. 5-9A is a comparison of zero-lag correlation $\rho[i]$ over the first and last stimulus cycles, for each stimulus velocity. As a measure of thalamic firing synchrony, $\rho[i]$ incorporates the internal dynamics of the neurons as well as network effects. The figure shows that the administration of the stimulus increases the correlation between the neurons at all velocities, and therefore changes the dependence. The figure also suggest that the change in dependence is more pronounced for the high and medium-velocity stimuli compared to the low-velocity stimulus. Moreover, there do not seem to be major differences between the first and last stimulus cycles. Fig. 5-10A is a

comparison of zero-lag correlation $\rho[i]$ across stimuli over the first and last stimulus cycles. The figure suggests that increases in correlation/dependence are stronger (and occur earlier with respect to the stimulus onset) for the high and medium velocity stimuli compared to the low-velocity stimulus. We also observe that these increases in correlation/dependence mirror changes in the stimuli.

Fig. 5-11 plots the coefficients $\gamma_c^{(c')}$ representing the effect of the history of Neuron c' on Neuron c $(c, c' = 1, 2)$.

Effect of Neuron 1 on itself (Fig. 5-11*A*, Column 1): the figure shows strong 1 ms inhibitory effects followed by milder excitatory behavior at 2 to 3 ms time scale. The high and medium velocity stimuli do not seem to exhibit major effects at longer time scales. However, the low-velocity stimulus appears slightly inhibitory from 5 to 25 ms.

Effect of Neuron 2 on Neuron 1 (Fig. 5-11*A*, Column 2): the history of Neuron 2 does not seem to have major effects on Neuron 1's present for the high and low-velocity stimuli. For the medium-velocity stimulus, the effect of Neuron 2's history on Neuron 1 oscillates between excitatory and inhibitory effects.

Effect of Neuron 1 on Neuron 2 (Fig. 5-11*B*, Column 1): The immediate history of Neuron 1 appears not to have any major effects on the present of Neuron 1 at the high and medium velocities. The low-velocity stimulus shows excitatory behavior at the 20 ms time scale, and inhibitory ones at the 40 ms time scale.

Effect of Neuron 2 on itself (Fig. 5-11*B*, Column 2): As in the case of Neuron 1, we see a strong initial inhibitory effect of Neuron 2's history on its present, at all velocities.

5.4.2 Summarizing results of analyses on all pairs

We computed the three measures of thalamic firing synchrony described previously for each pair of neuron and stimulus velocity. For each stimulus velocity, we took the median of these quantities as a summary over the population.

Fig. 5-12*A* compares the modulation of the non-simultaneous and simultaneous events by all three stimuli, across the population. The figure shows that the high and medium-velocity stimuli modulate each of the simultaneous and non-simultaneous

events. The low-velocity stimulus modulates the non-simultaneous events to some extent but not the simultaneous event. The figure also suggests that the stimulus modulation of non-simultaneous and simultaneous events is similar for the high and medium-velocity stimuli. Moreover, for both these stimuli, the modulation of the simultaneous event is much stronger than that of the non-simultaneous events. In short, across the population, the stimulus induces zero-lag thalamic firing synchrony for the high and medium-velocity stimuli but not the low-velocity stimulus. This is more apparent from Fig. 5-14*A*, which compares the stimulus modulation of the '11' event across stimuli. This figure also suggests that the maximum stimulus modulation of the simultaneous event occurs earlier with respect to the stimulus onset for the high-velocity stimulus, compared to the medium-velocity stimulus. Fig. 5-13 displays the empirical distribution of the time of occurrence of maximum stimulus modulation with respect to the stimulus onset. The figure shows that the the higher the stimulus velocity, the earlier the time of maximum stimulus modulation of the simultaneous '11' event with respect to the stimulus onset. Moreover, it appears that the time of occurrence of maximum stimulus modulation is more robust across the population for high and medium-velocity stimuli (Table 5.1).

	Stim 1	Stim 2	Stim 3
μ	12.8	19.6	57.2
σ	2.1	3.2	42.5

Table 5.1. Second-order statistics of data in Fig. 5-13.

Fig. 5-15*A* compares zero-lag correlation $\rho[i]$ across the population over the first and last stimulus cycles. for each stimulus velocity. The figure shows that the administration of the stimulus increases the correlation between the neurons at high and medium velocities, and therefore changes the dependence. The change in dependence is more pronounced for the high and medium-velocity stimuli compared to the low-velocity stimulus. There do not seem to be major differences between the first and last stimulus cycles. Fig. 5-16*A* is a comparison of zero-lag correlation $\rho[i]$ across stimuli. The figure suggests that increases in correlation/dependence are stronger (and occur earlier with respect to the stimulus onset) for the high and medium-

velocity stimuli compared to the low-velocity stimulus. Moreover, these increases in correlation/dependence mirror changes in the stimuli. We also observe that the peak correlation occurs earlier, with respect to the stimulus onset, for the high-velocity stimulus compared to the medium-velocity one.

Figure 5-17 plots the coefficients representing the effect of the history of the neurons in a pair for the whole population. Across the population, each neuron in a pair shows initial 1 to 2 ms refractory effects at all velocities. There also appear to be mild excitatory cross effects of each neuron on the other neuron in the pair at the 1 to 2 ms time scale.

5.5 Decoding examples

In this section, we use the results of the analyses above and the data not used for training (test data), to decode the stimuli. In other words, for each stimulus, we treat the parameters of the mGLMs as ground truth, and use the test data for those pairs to form an estimate of the stimulus. We use 11 of the 17 pairs in our data set, whose raster plots clearly show the effect of the stimulus on the joint spiking activity of the pairs.

We recall that, in the GLM analyses of the previous section, reducing the AIC-optimal values of J to $J = 2$ did not significantly increase the likelihood, nor did it worsen the goodness-of-fit as measured by KS plots. So, in what follows, we use the same value of $J = 2$ for all pairs of neurons.

In our decoding set-up of Chapter 4, we assume that the state $\theta_i = (s_i, s_{i-1})$ and that it follows the random walk of Equation 4.31, with $F_i = \mathrm{I}$ and $\Sigma_i = \sigma\mathrm{I}$. Conditioned on the state, we assume that the 11 pairs are independent and that, for a given pair, trials are independent. This leads to the following decoding algorithm

$$\theta_{i|i-1} = \theta_{i-1|i-1}, \tag{5.4}$$

$$W_{i|i-1} = W_{i-1|i-1} + \Sigma_i, \tag{5.5}$$

$$W_{i|i}^{-1} = W_{i|i-1}^{-1} + \sum_{p=1}^{11} \beta_p^{(0)} \cdot \sum_{r=1}^{33} \left[\left(\text{diag } \lambda_{i,r,p}^* \Delta - \lambda_{i,r,p}^* \lambda_{i,r,p}^{*\prime} \Delta^2 \right) \right]_{\theta_{i|i-1}} \cdot \beta_p^{(0)\prime} \quad (5.6)$$

$$\theta_{i|i} = \theta_{i|i-1} + W_{i|i} \sum_{p=1}^{11} \beta_p^{(0)} \sum_{r=1}^{33} \left[\left(\Delta N_{i,r,p}^* - \lambda_{i,r,p}^* \Delta \right) \right]_{\theta_{i|i-1}}, \quad (5.7)$$

where p and r are the indices over pairs and trials respectively and $\beta_p^{(0)} = [\beta_{1,p}^{(0)} \beta_{2,p}^{(0)} \beta_{3,p}^{(0)}]$, is the 2-by-3 matrix whose m^{th} column is the vector of mGLM coefficients corresponding to the stimulus effect on the m^{th} component (Equation 5.1), i.e. $\beta_{m,p}^{(0)} = (\beta_{m,0,p}^{(0)}, \cdots, \beta_{m,J-1,p}^{(0)})'$. The index p indicates that the stimulus effect is different for different pairs of neurons.

We compare the decoding algorithm above to one based on a model which assumes that the neurons in each pair are independent. The resulting algorithm is

$$\theta_{i|i-1} = \theta_{i-1|i-1}, \quad (5.8)$$

$$W_{i|i-1} = W_{i-1|i-1} + \Sigma_i, \quad (5.9)$$

$$W_{i|i}^{-1} = W_{i|i-1}^{-1} + \sum_{p=1}^{11} \sum_{c=1}^{2} \beta_{c,p}^{(0)} \cdot \sum_{r=1}^{33} \left[\lambda_{i,r,p,c} \Delta (1 - \lambda_{i,r,p,c} \Delta) \right]_{\theta_{i|i-1}} \cdot \beta_{c,p}^{(0)\prime} \quad (5.10)$$

$$\theta_{i|i} = \theta_{i|i-1} + W_{i|i} \sum_{p=1}^{11} \sum_{c=1}^{2} \beta_{c,p}^{(0)} \sum_{r=1}^{33} \left[\left(\Delta N_{i,r,p,c} - \lambda_{i,r,p,c} \Delta \right) \right]_{\theta_{i|i-1}}, \quad (5.11)$$

where c is the index for neurons in a pair (which we assume are independent), and $\beta_{c,p}^{(0)} = (\beta_{c,0,p}^{(0)}, \cdots, \beta_{c,J-1,p}^{(0)})'$ is the vector of GLM coefficients corresponding to the stimulus effect on the c^{th} neuron of pair p.

5.5.1 Decoding results on real data

Fig. 5-18 compares the decoded low-velocity stimulus using independent and joint decoding to the waveform programmed into the mechanical device responsible for whisker motion. The figure shows that the stimuli decoded using either methods are very similar and resemble the ideal, periodic stimulus. In terms of mean-squared error (MSE), the stimulus obtained using the joint model is closer to the administered stimulus. To highlight differences, Fig. 5-19 compares the algorithms over the first

and last cycles, as well as the averages (over the 16 cycles) of the decoded waveforms. All three panels of the figure indicate that, in each cycle, the low-velocity stimulus comprises of two successive deflections. This would explain the two distinct peaks in the correlation plot for the low-velocity stimulus (Fig. 5-9A, 3rd Column). Moreover, in Fig. 5-9A, 3rd Column, the 2nd peak is stronger over the last cycle (black trace). This could be explained by the difference in the decoded stimulus over the 1st cycle (Fig. 5-19A) and the last cycle (Fig. 5-19B). Indeed, the decoded secondary deflection is smaller in the 1st cycle compared to the last cycle. One could argue that the observations of Fig. 5-9 apply to one pair only, whose contribution to the decoding algorithm may have (somehow) skewed the decoding results. We removed this pair and others (one at a time) from the decoding algorithms and obtained traces nearly identical to Figs. 5-18 and 5-19. We are able to obtain plots similar to Fig. 5-18 for the medium and high-velocity stimuli. In both cases, the stimuli decoded show features similar to those of Fig. 5-18, such as the periodicity of the decoded waveform. However, the presence in each cycle of two successive deflections, as well as the difference (noted above) between the first cycle and the last cycles (Fig. 5-19A and B) are unique to the low-velocity stimulus. Figs. 5-21 and 5-20 compare the cycle-average of the decoded stimulus to one cycle of the waveforms programmed into the mechanical device responsible for whisker motion. We focus on the medium and high-velocity stimuli as we have discussed the low-velocity stimulus above in detail. Fig. 5-21 shows that the decoded medium and high-velocity stimuli are close to the administered stimulus in the regions where the stimuli are non-zero (0 to $\approx$ 25 ms and 0 to $\approx$ 40 ms, respectively). However, there is a discrepancy between the two in the regions where the administered stimuli are zero. This can be attributed to our stochastic continuity constraint (Eq. 4.31), which does not allow for sharp changes in the value of the decoded signal and/or noise when going from the ideal stimulus to the movement of the whisker.

Should we treat the available stimulus as ground truth?

The desired periodic stimuli were administered to the whisker using a piezoelectric stimulator [42]. Our mGLM analyses have assumed a one-to-one correspondence between the administered, ideal, periodic stimuli and whisker movement. In other words, we assumed the absence of errors/noise in going from the stimuli to the movement of the whisker, and used the ideal stimuli as inputs to our mGLM fits (Eq. 5.1). These errors could be due to imperfections in the placement of the whisker during the administration of the stimulus. Figure 5-21 shows that the decoded stimuli resemble the administered, ideal stimuli, especially at high and medium velocity. However, there are discrepancies, notably at low velocity. The presence of the secondary deflection is particularly puzzling.

Using simulated data, we study whether the discrepancies between the administered and the decoded stimuli are an artifact of the decoding algorithm. If this is not the case, then these discrepancies could be attributed to (a) inaccuracies in our model, which is doubtful given the goodness-of-fit results, or (b) noise in the stimuli delivered using the piezoelectrode: in other words, contrary to our assumptions, the administered whisker movement is *not* transferred exactly to the whisker. This could be addressed by explicitly accounting for errors in the stimulus in Eq. 5.1.

5.5.2 Decoding results on simulated data

Figs. 5-22 and 5-23 show the result of decoding the administered stimuli using simulated data. The leftmost panel of Fig. 5-23 shows the result of decoding the low-velocity stimulus. There are two important observations to make. First, the decoded stimulus is nearly identical to the ideal stimulus used in the simulation. This is a textbook example of the usefulness of the SEMPP decoding algorithms introduced in the previous chapter. Second, we notice the absence of the secondary deflection present in the third panel of Fig. 5-21. This leads us to the conclusion that the two successive deflections are unlikely to be an artifact of the decoding algorithm. Fig. 5-18 may very well constitute an accurate estimate of the actual motion of the whisker

during the experiment. This estimate of the low-velocity stimulus is characterized by (a) the presence of two successive deflections in each cycle, and (b) a different form of the stimulus in the 1st cycle when compared to the last cycle (Fig. 5-19), which is similar to the other 14 cycles.

We also note in Fig. 5-23 that, while preserving their overall shape, the decoding algorithm slightly underestimates the medium and high-velocity stimuli. This could be due to inaccuracies in the implementation of the algorithm used to simulate the data. It is also possible that the decoding algorithm is not able to track the fast changes in the high and medium velocity stimuli around their peak values.

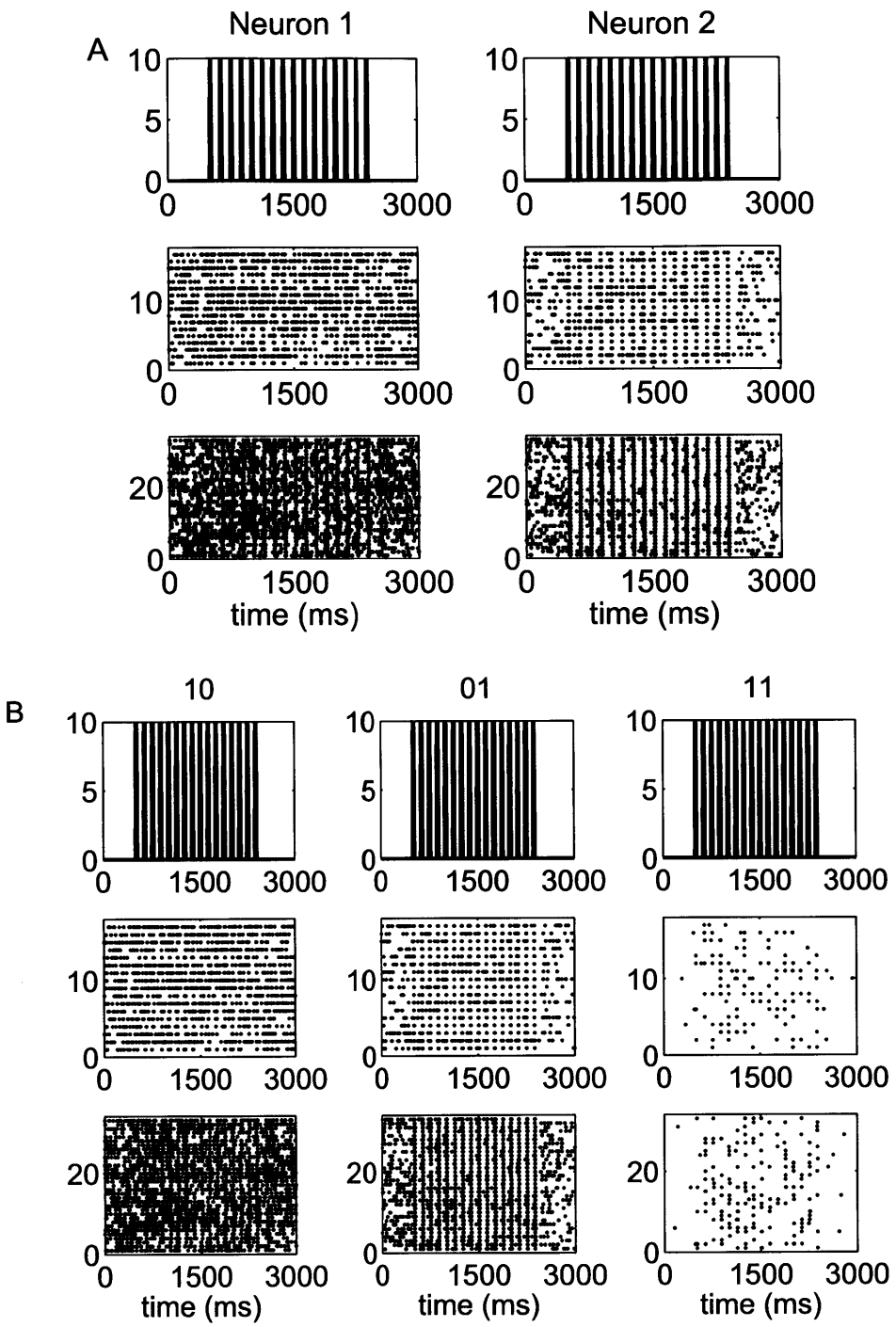

Figure 5-1. Raster plots of the spiking activity of a representative pair of neurons in response to a periodic whisker deflection of velocity $v = 80$ mm/s. (A) Standard raster plots, (B) New raster plots of the *joint* events, '01', '10' and '11'. In both cases, the first row displays the stimulus, while the second and third rows display the training and test sets respectively. The standard raster plots (A) show that the stimulus induces strong modulation of the neural spiking of each of the neurons. These standard raster plots do not show the effect of the stimulus on joint spiking. The new raster plots (B) show a modulation of the joint spiking activity ('11') by the stimulus.

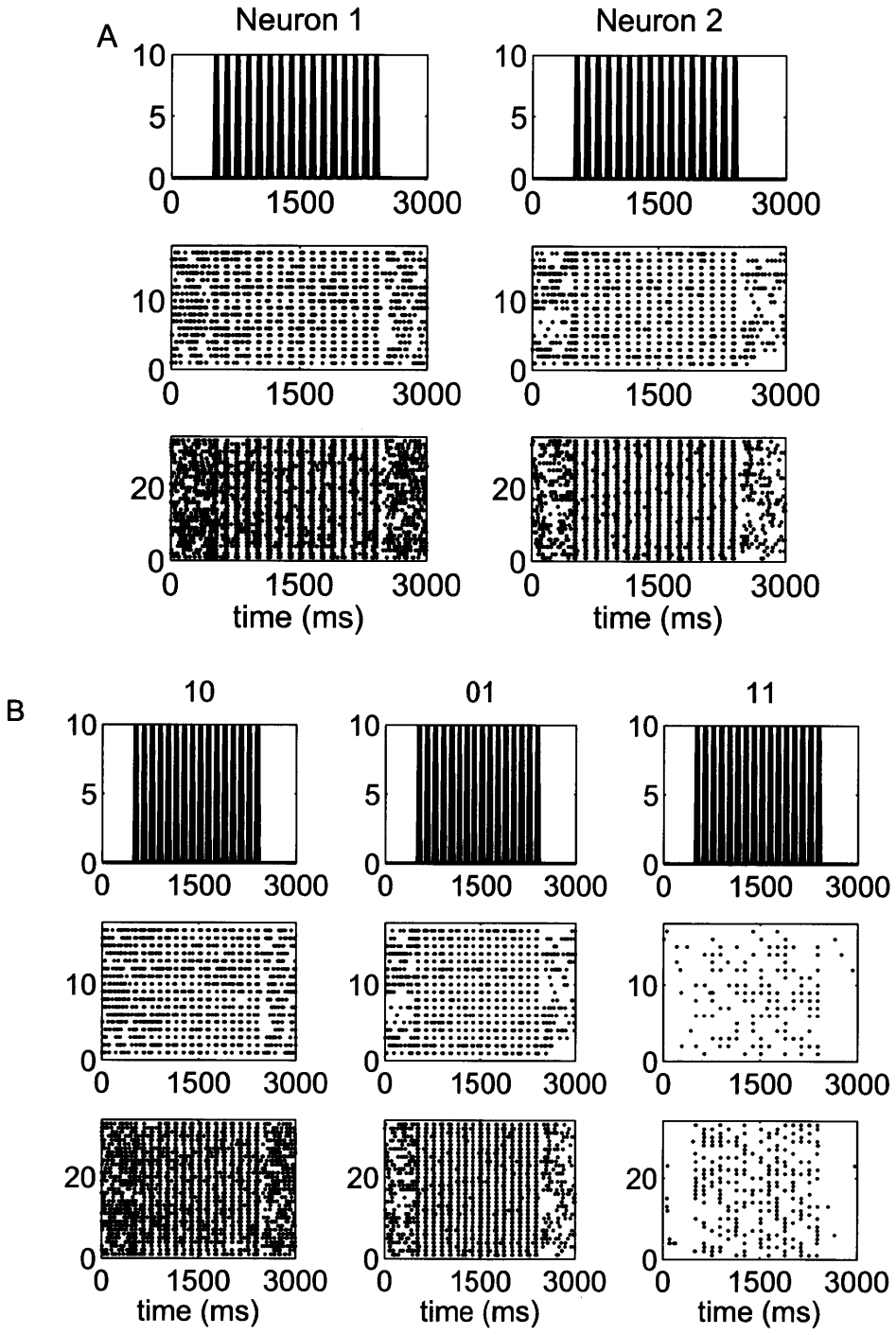

Figure 5-2. Raster plots of the spiking activity of a representative pair of neurons in response to a periodic whisker deflection of velocity $v = 50$ mm/s. (A) Standard raster plots, (B) New raster plots of each of the *joint* events, '01', '10' and '11'. In both cases, the first row displays the stimulus, while the second and third rows display the training and test sets respectively. The standard raster plots (A) show that the stimulus induces strong modulation of the neural spiking of each of the neurons. These standard raster plots do not show the effect of the stimulus on joint spiking. The new raster plots (B) show a modulation of the joint spiking activity ('11') by the stimulus.

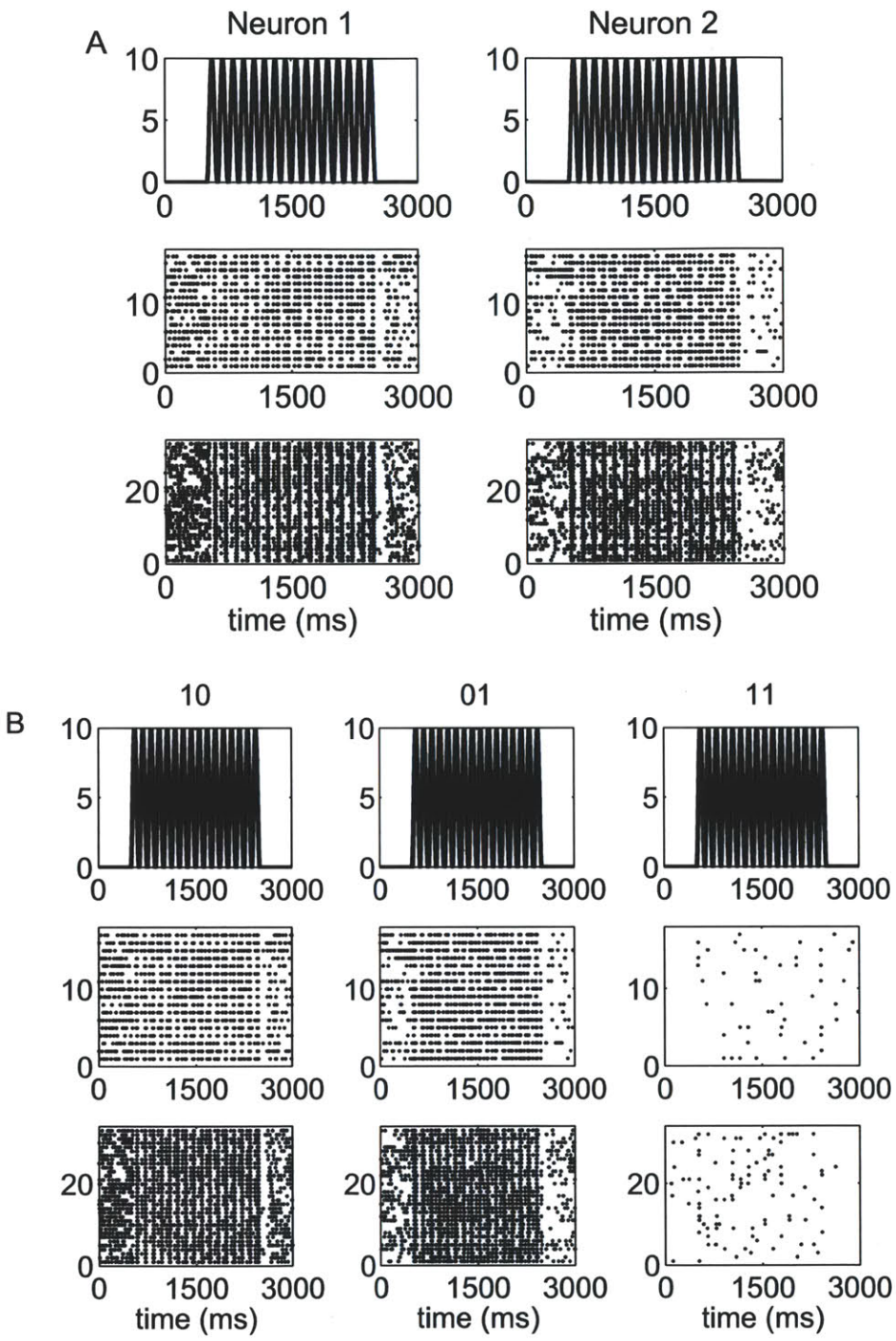

Figure 5-3. Raster plots of the spiking activity of a representative pair of neurons in response to a periodic whisker deflection of velocity $v = 16$ mm/s. (A) Standard raster plots, (B) New raster plots of each of the *joint* events, '01', '10' and '11'. In both cases, the first row displays the stimulus, while the second and third rows display the training and test sets respectively. The standard raster plots (A) show that the stimulus induces strong modulation of the neural spiking of each of the neurons. These standard raster plots do not show the effect of the stimulus on joint spiking. The new raster plots (B) show a modulation of the joint spiking activity ('11') by the stimulus.

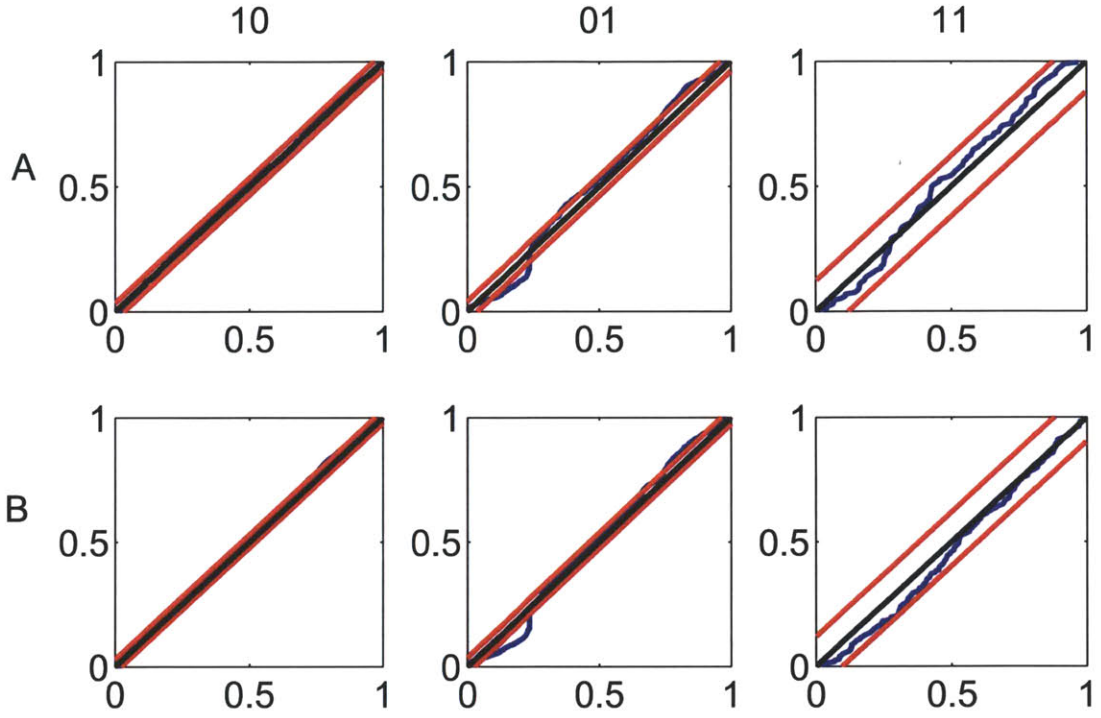

Figure 5-4. Goodness-of-fit assessment by KS plots based on the time-rescaling theorem for the pair in Fig. 5-1. (A) Time-rescaling performance on the training data. (B) Time-rescaling performance on the test data. In both cases, the parallel red lines correspond to the 95% confidence bounds. The KS plots show that the model fits both the training and test data well. The good KS performance on each of the components of ΔN^* demonstrates the model's accurate description of the *joint* process. The performance on the test data demonstrates the strong predictive power of the model.

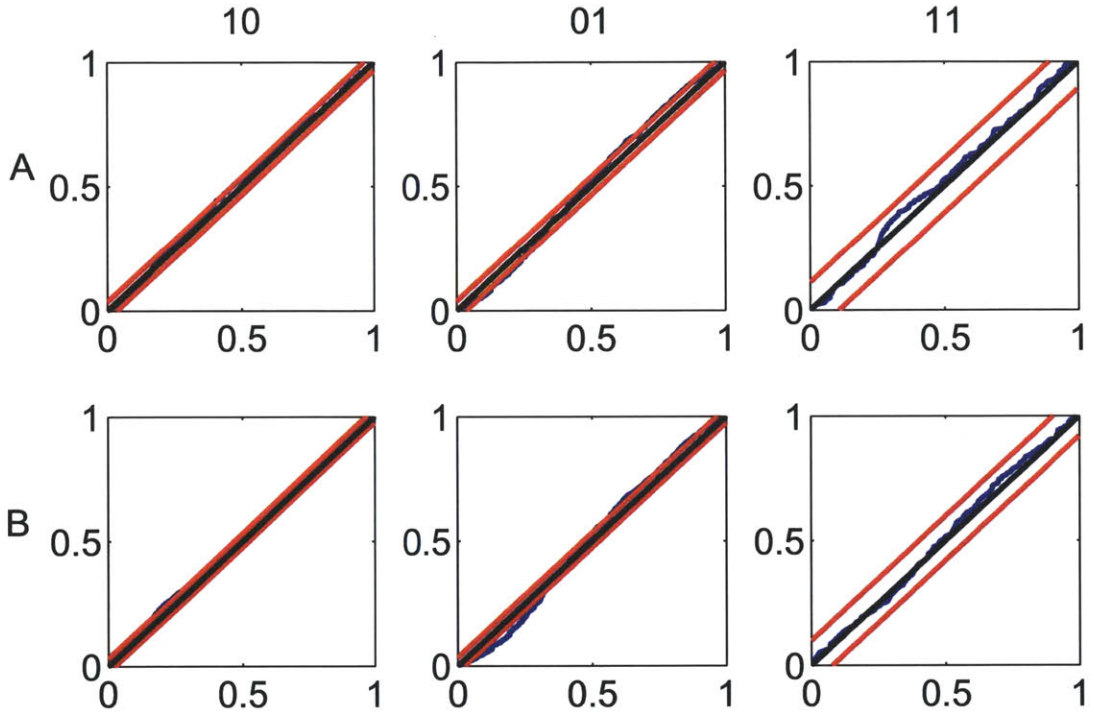

Figure 5-5. Goodness-of-fit assessment by KS plots based on the time-rescaling theorem for the pair in Fig. 5-2. (A) Time-rescaling performance on the training data. (B) Time-rescaling performance on the test data. In both cases, the parallel red lines correspond to the 95% confidence bounds. The KS plots show that the model fits both the training and test data well. The good KS performance on each of the components of ΔN^* demonstrates the model's accurate description of the *joint* process. The performance on the test data demonstrates the strong predictive power of the model.

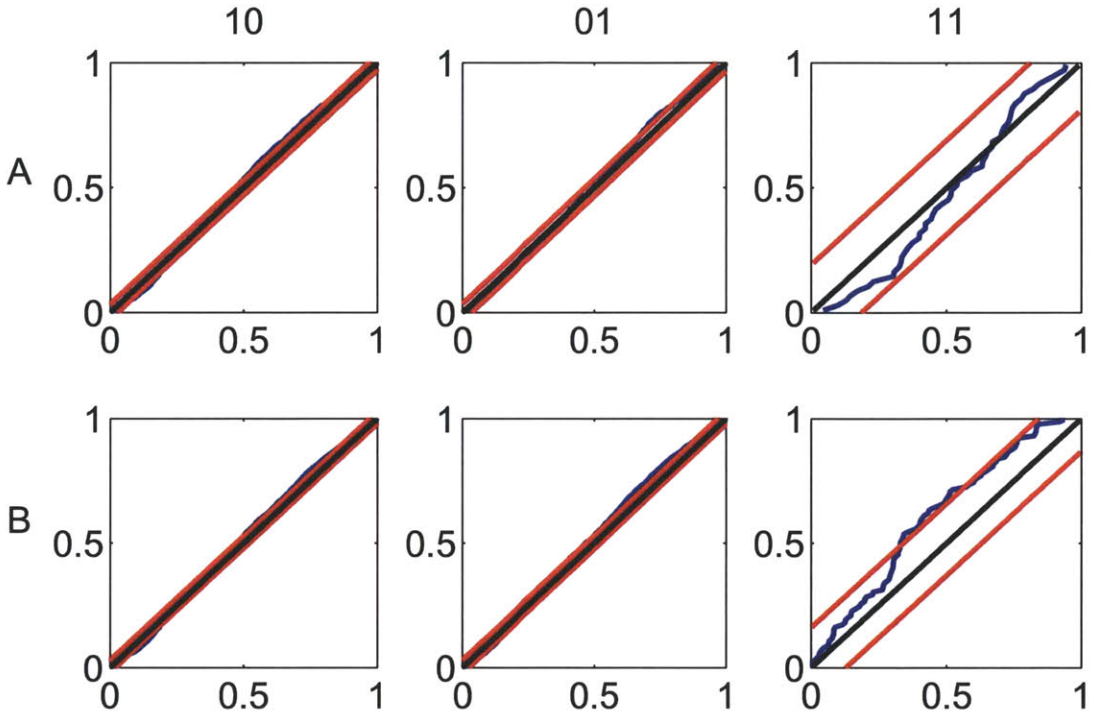

Figure 5-6. Goodness-of-fit assessment by KS plots based on the time-rescaling theorem for the pair in Fig. 5-3. (A) Time-rescaling performance on the training data. (B) Time-rescaling performance on the test data. In both cases, the parallel red lines correspond to the 95% confidence bounds. The KS plots show that the model fits both the training and test data well. The good KS performance on each of the components of ΔN^* demonstrates the model's accurate description of the *joint* process. The performance on the test data demonstrates the strong predictive power of the model.

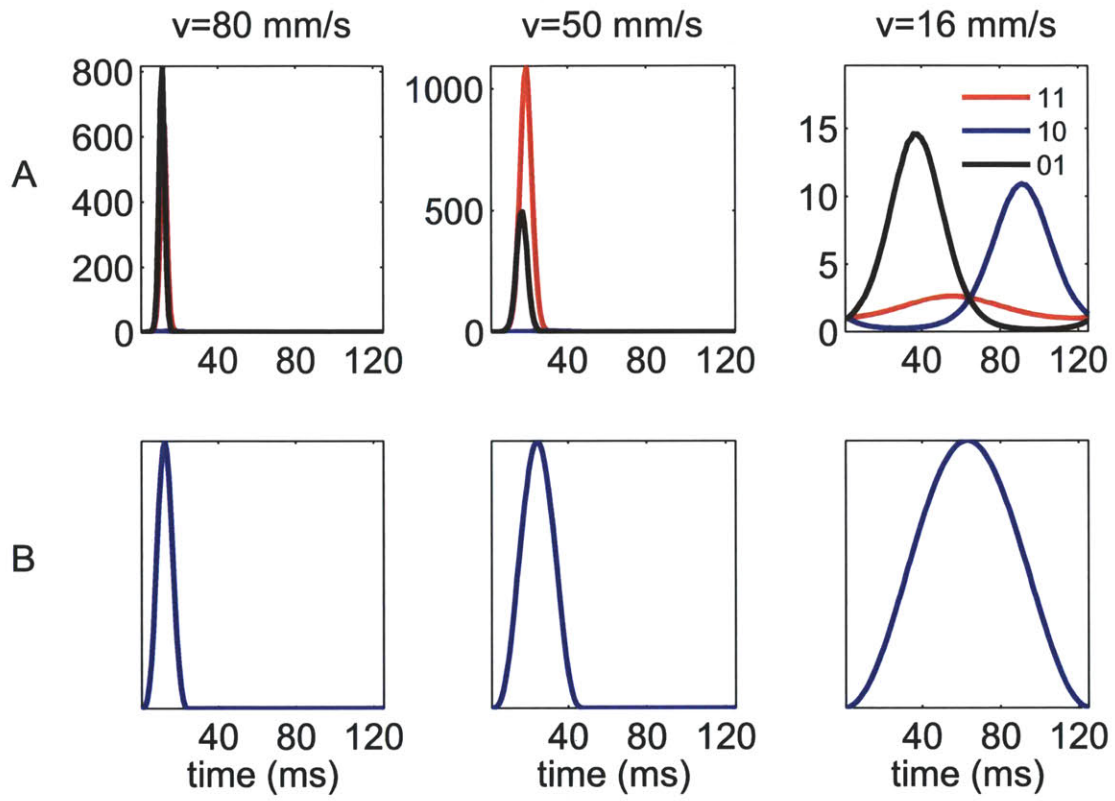

Figure 5-7. Comparison of the modulation of non-simultaneous and simultaneous events for each stimulus velocity. (A) Stimulus modulation, (B) Stimulus over a single cycle. The figure shows that, for each stimulus velocity, the stimulus modulates all of the joint events. For this pair, there is strong stimulus-induced thalamic firing synchrony for the high and medium-velocity stimuli, as measured by the stimulus modulation of the '11' event. For the said stimuli, the stimulus modulation of the '11' event is on the same order as that of the '01' event and much stronger than that of the '10' event. There is evidence of stimulus-induced thalamic firing synchrony for the low-velocity stimulus, albeit to a much lower extent that for the other stimuli.

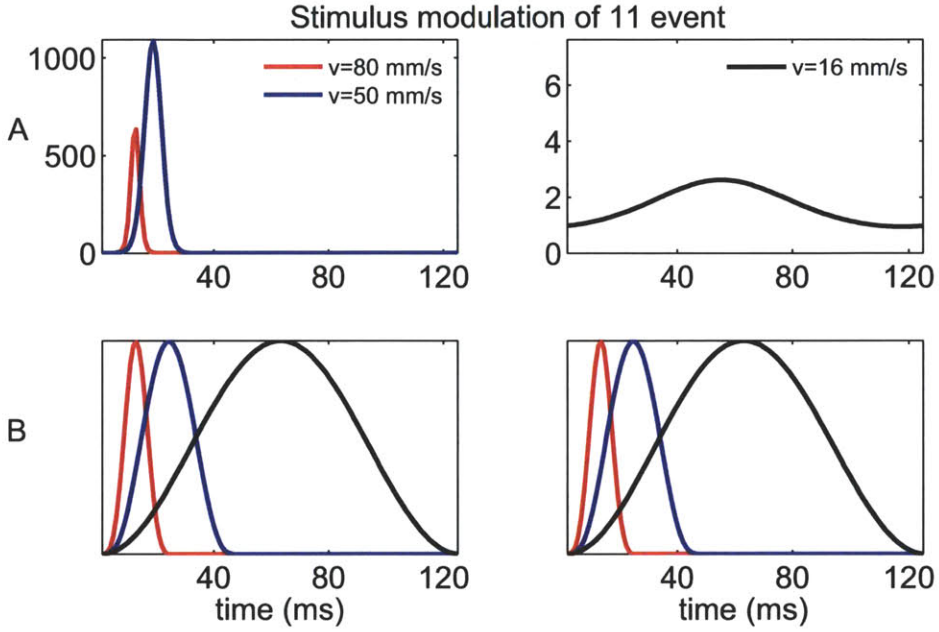

Figure 5-8. Comparison of the modulation of the simultaneous '11' event across stimuli. (A) Stimulus modulation of '11' event for all three stimuli. (B) Stimuli over a single cycle. For this pair, zero-lag stimulus-induced thalamic firing synchrony, as measured by the stimulus modulation of the '11 event, is two orders of magnitude stronger for the high and medium-velocity stimuli compared to the low-velocity stimulus.

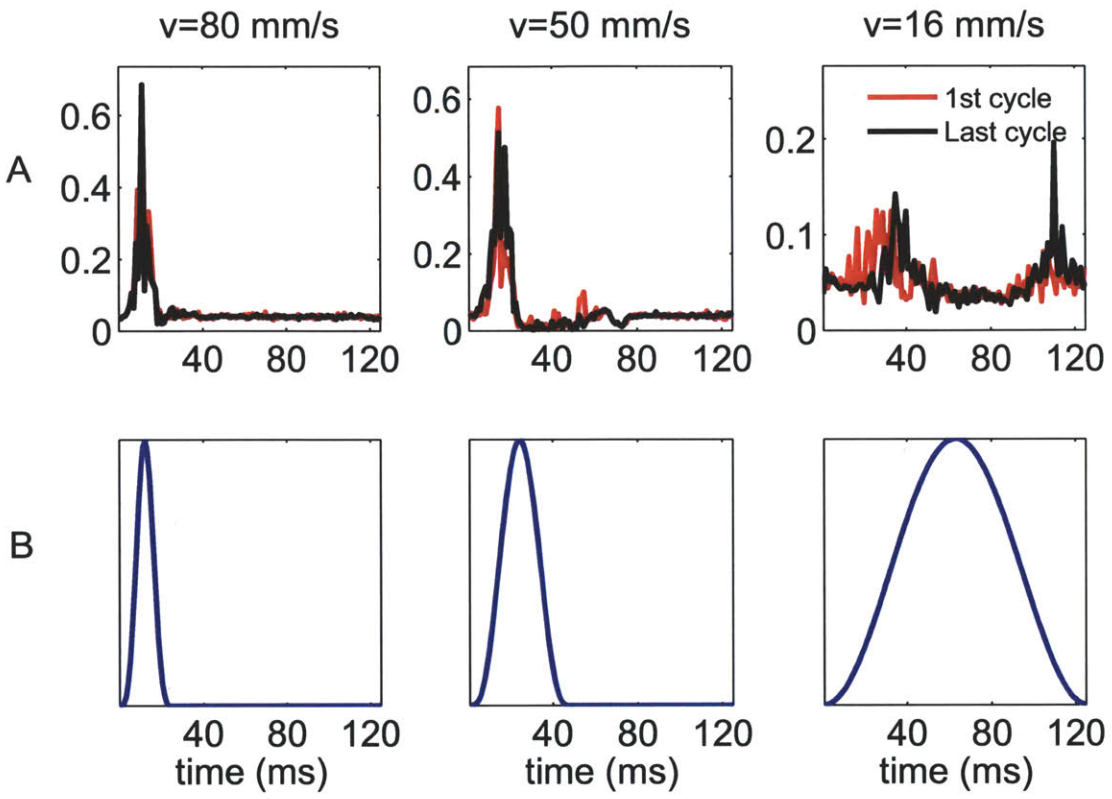

Figure 5-9. Comparison of zero-lag correlation $\rho[i]$ over the first and last stimulus cycles, for each stimulus velocity. (A) Zero-lag correlation $\rho[i]$ over first and last cycles, for each stimulus, (B) Stimulus over a single cycle. This measure of zero-lag dependence takes into account the internal dynamics of the neurons as well as network effects. The figure shows that the administration of the stimulus increases the correlation between the neurons at all velocities, and therefore changes the dependence. The figure also suggests that the change in dependence is more pronounced for the high and medium-velocity stimuli compared to the low-velocity stimulus. Moreover, there do not seem to be major differences between the first and last stimulus cycles.

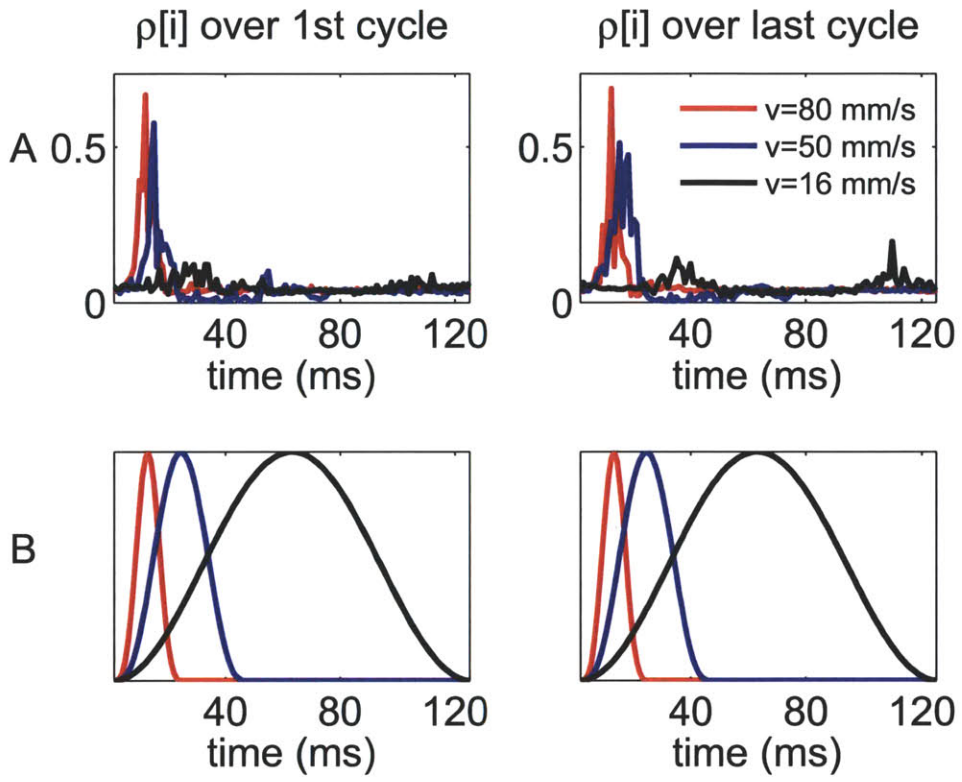

Figure 5-10. Comparison of zero-lag correlation $\rho[i]$ across stimuli over the first and last stimulus cycles. (A) Zero-lag correlation $\rho[i]$ over first and last cycles, (B) Stimulus over a single cycle. This figure confirms our observation from Figure 5-9 that increases in correlation/dependence are stronger for the high and medium velocity stimuli compared to the low-velocity stimulus. Moreover, changes in the dependence mirror changes in the stimuli at high and medium velocities.

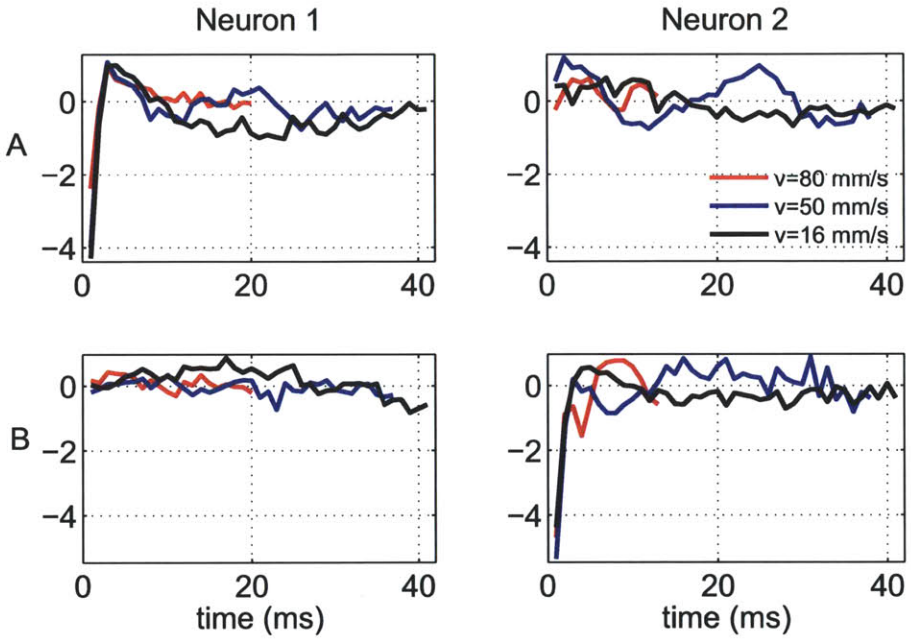

Figure 5-11. Effect of the history of each neuron in the pair on its own firing and on the other neuron's firing. (A) History effect on Neuron 1's firing, (B) History effect on Neuron 2's firing. The first and second columns represent the effects of Neuron 1 and 2 respectively. Both neurons show initial 1 to 2 ms refractory effects at all velocities. Neuron 2 shows mild excitatory effects on Neuron 1 for the medium-velocity stimulus. More details can be found in the text.

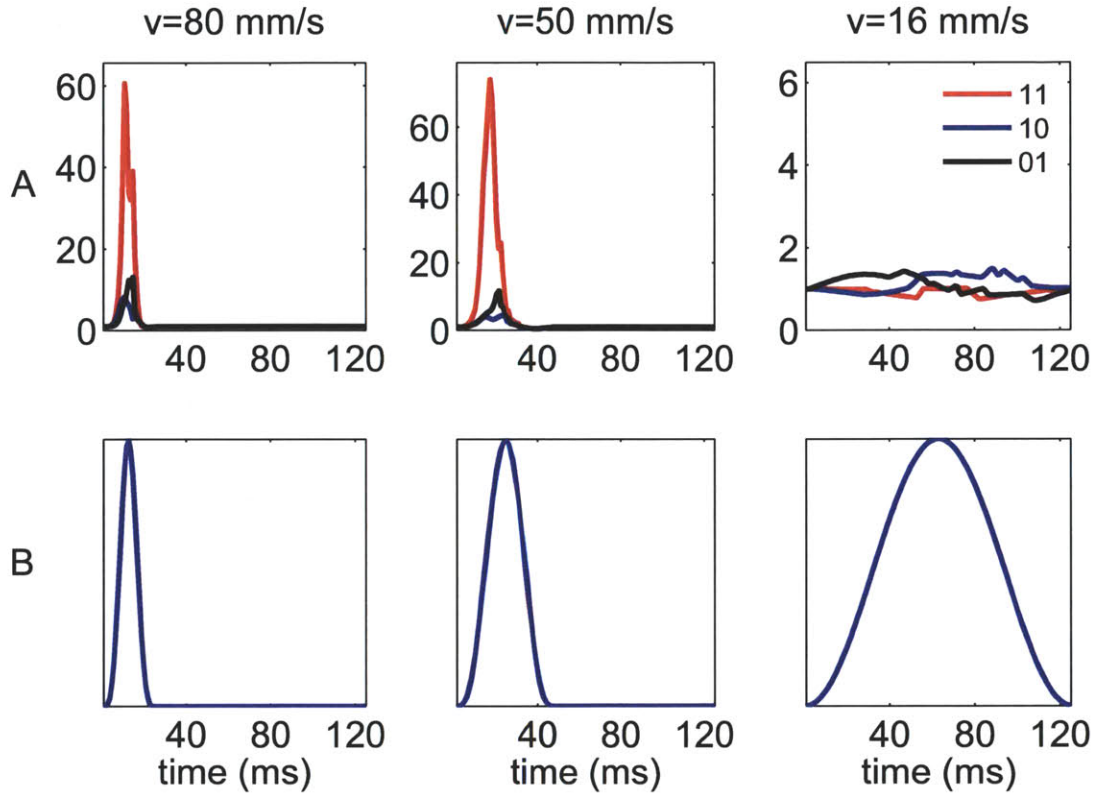

Figure 5-12. Population comparison of the modulation of non-simultaneous and simultaneous events for each stimulus velocity. (A) Stimulus modulation, (B) Stimulus over a single cycle. The figure shows that, for each stimulus velocity, the stimulus modulates all of the joint events across the population. There is strong stimulus-induced thalamic firing synchrony for the high and medium-velocity stimuli, as measured by the stimulus modulation of the '11' event. For the said stimuli, the stimulus modulation of the '11' event across the population is stronger than that of the '10' and '01' events. There is no strong evidence of stimulus-induced thalamic firing synchrony for the low-velocity stimulus.

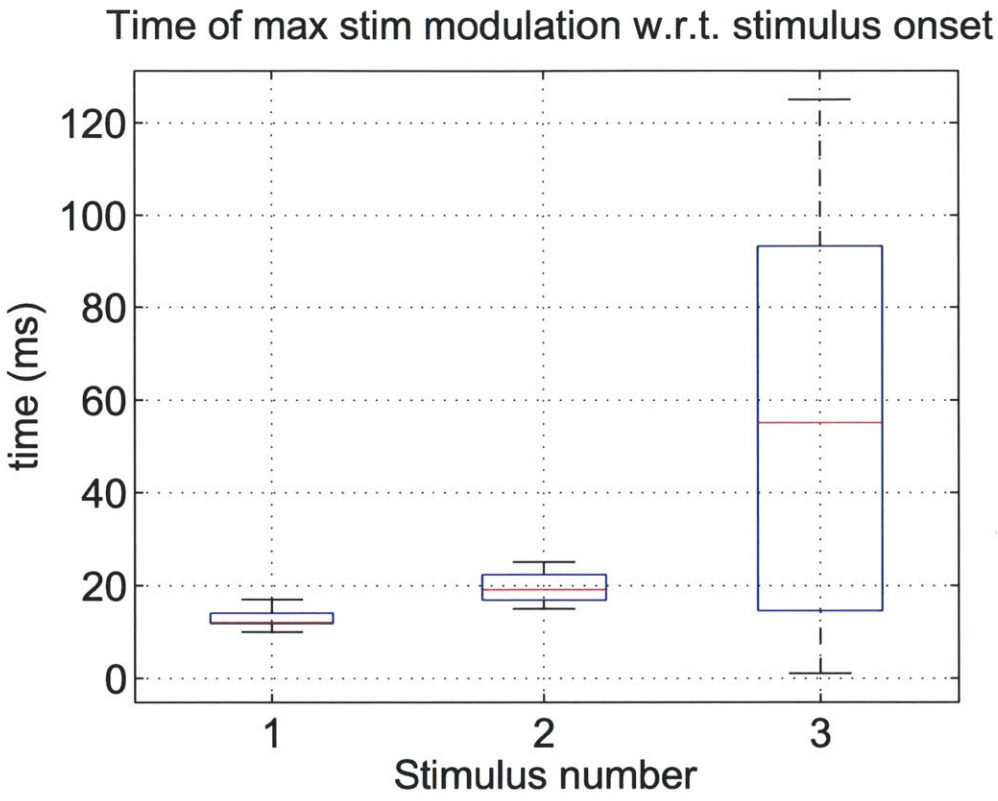

Figure 5-13. Empirical distribution of the time of occurrence of maximum stimulus modulation with respect to stimulus onset for all 17 pairs in the data set. The figure suggests that, the higher the stimulus velocity, the earlier the time of maximum stimulus modulation of the simultaneous '11' event with respect to the stimulus onset. Moreover, it appears that the time of occurrence of maximum stimulus modulation is more robust across the population for high and medium-velocity stimuli. See Table

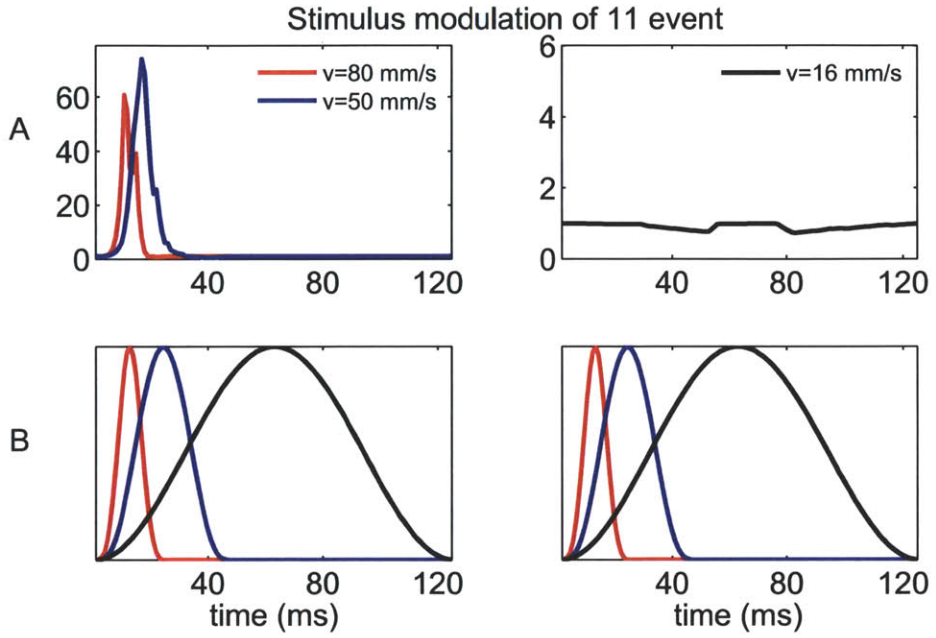

Figure 5-14. Population comparison of the modulation of the simultaneous '11' event across stimuli. (A) Stimulus modulation of '11' event for all three stimuli. (B) Stimuli over a single cycle. For this pair, zero-lag stimulus-induced thalamic firing synchrony, as measured by the stimulus modulation of the '11 event, is two orders of magnitude stronger for the high and medium-velocity stimuli compared to the low-velocity stimulus.

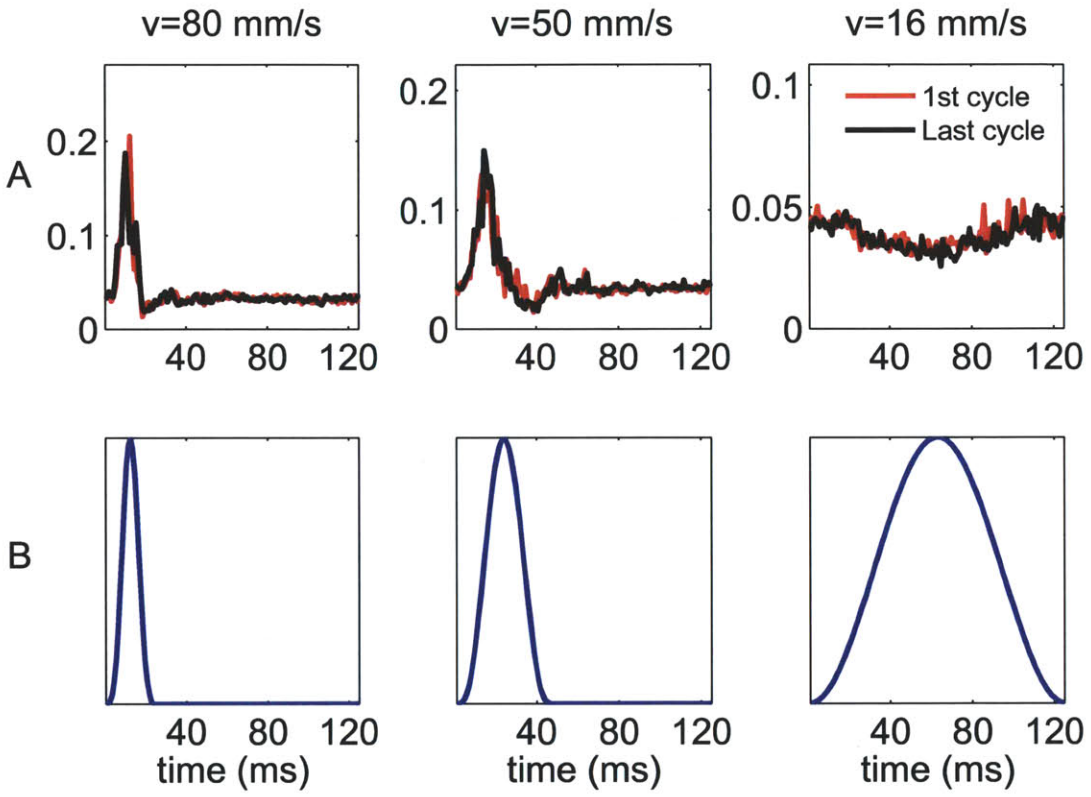

Figure 5-15. Population comparison of zero-lag correlation $\rho[i]$ over the first and last stimulus cycles. (A) Zero-lag correlation $\rho[i]$ over first and last cycles, (B) Stimulus over a single cycle. This measure of zero-lag dependence takes into account the internal dynamics of the neurons as well as network effects. The figure shows that, across the population, the administration of the stimulus increases the correlation between the neurons at high and medium velocities, and therefore changes the dependence for those stimuli. The change in dependence is more pronounced for the high velocity stimulus compared to the medium-velocity stimulus. For the low-velocity stimulus, there is no evidence of changes in dependence across the population. Lastly, the figure suggests that there are no major differences between the first and last stimulus cycles.

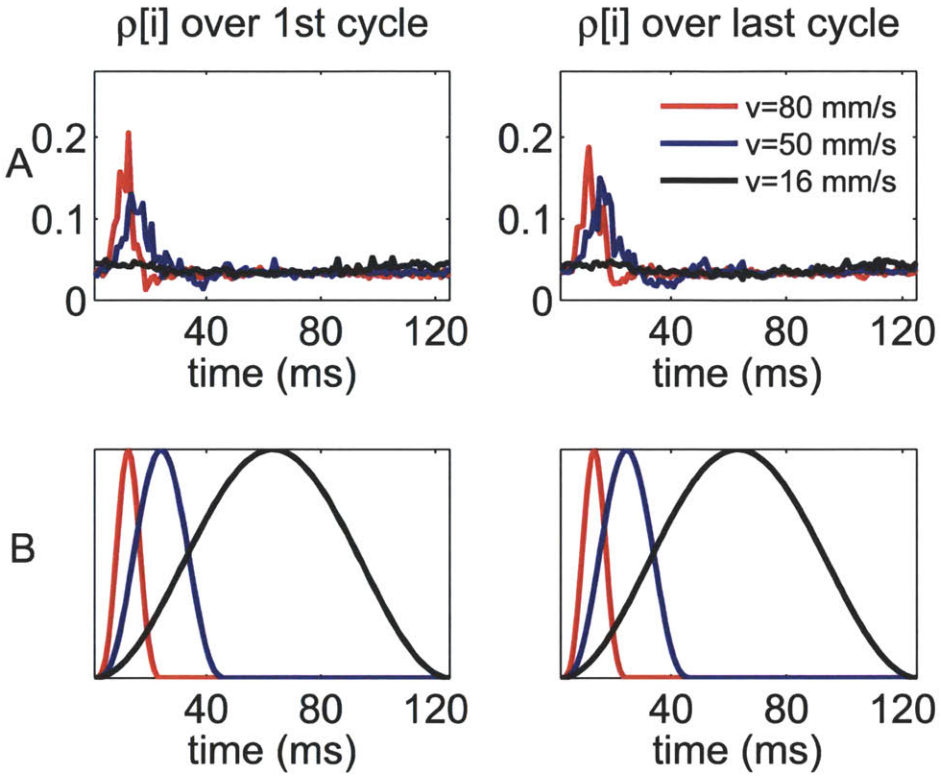

Figure 5-16. Population comparison of zero-lag correlation $\rho[i]$ across stimuli over the first and last stimulus cycles. (A) Zero-lag correlation $\rho[i]$ over first and last cycles, (B) Stimulus over a single cycle. The figure confirms our observation from Figure 5-15 that increases in correlation/dependence are strong for the high and medium-velocity stimuli but not for the low-velocity stimulus. Moreover, changes in the dependence mirror changes in the stimuli at high and medium velocities.

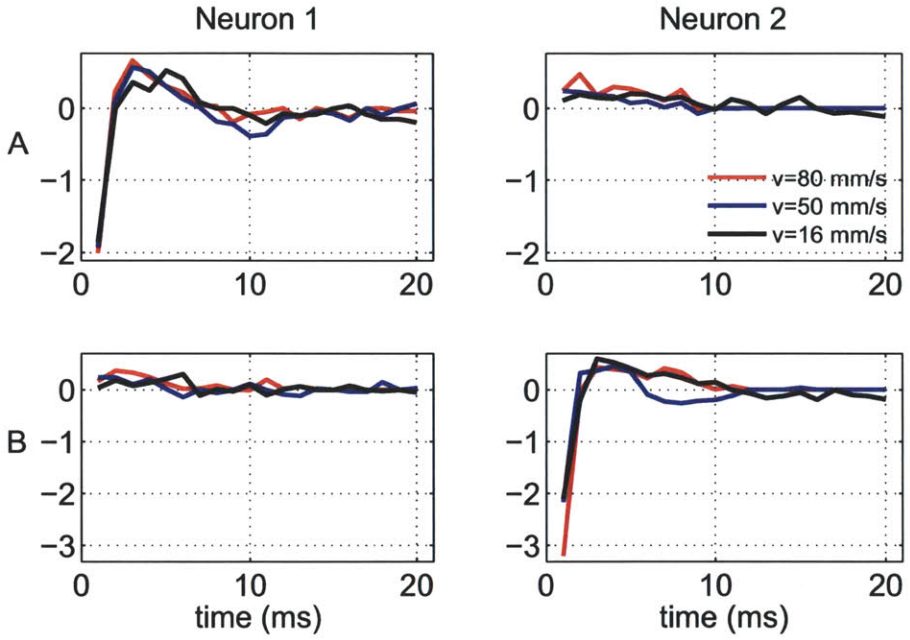

Figure 5-17. Population summary of each neuron's effect on its own firing and on the other neuron's firing. (A) Median history effect on Neuron 1's firing, (B) Median history effect on Neuron 2's firing. The first and second columns represent the effects of Neuron 1 and 2 respectively. Across the population, each neuron in a pair shows initial 1 to 2 ms refractory effects at all velocities. There also appear to be mild excitatory cross effects of each neuron on the other neuron in the pair.

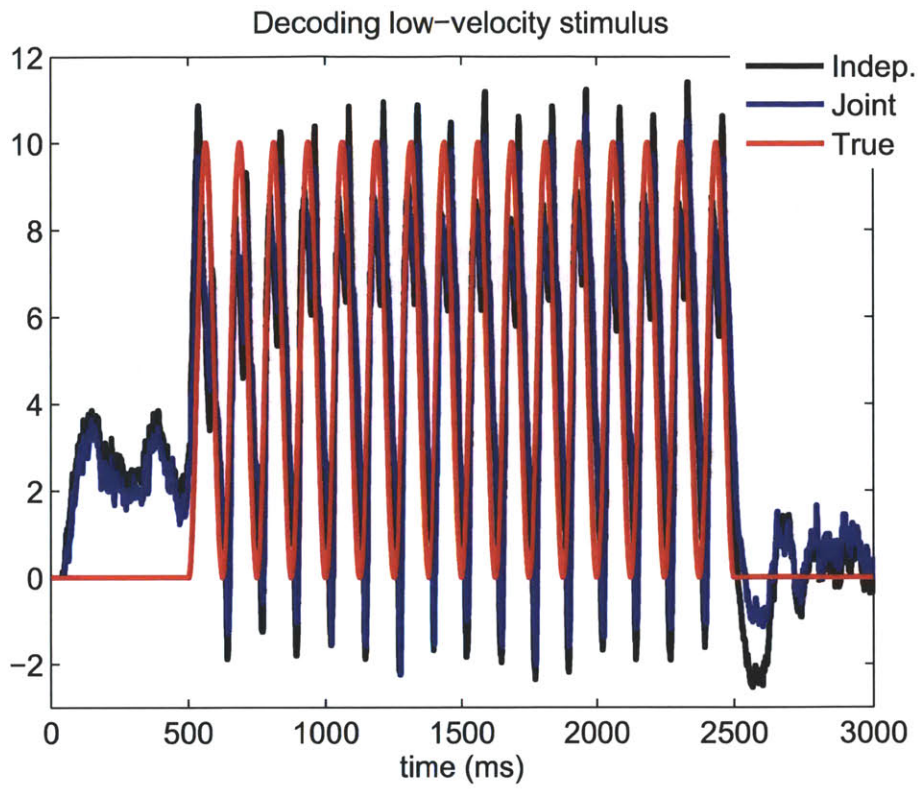

Figure 5-18. Decoded low-velocity stimulus using independent and joint decoding. The figure shows that the stimuli decoded using either methods are very similar and resemble the ideal, periodic stimulus. In terms of MSE, the stimulus obtained using the joint model is closer to the administered stimulus. To highlight differences, Fig. 5-19 shows a comparison over the first and last cycles, as well as averaged over cycles.

5.6 Summary of findings

We proposed a simultaneous-event multivariate point-process framework to characterize the joint dynamics of pairs of thalamic neurons in response to periodic whisker deflections varying in velocity. A multinomial GLM model of these data offered a very compact representation of the joint dynamics of the said neuronal pairs. The model uncovered history effects of the neurons on their joint firing propensity which lagged up to 40 ms in the past (Fig. 5-11). The advantage of this approach over existing point-process techniques is that it is able to model simultaneous occurrence of events. Its main advantage over histogram-based ones is its ability to relate the joint spiking propensity of neurons to stimuli as well as the history of the neurons.

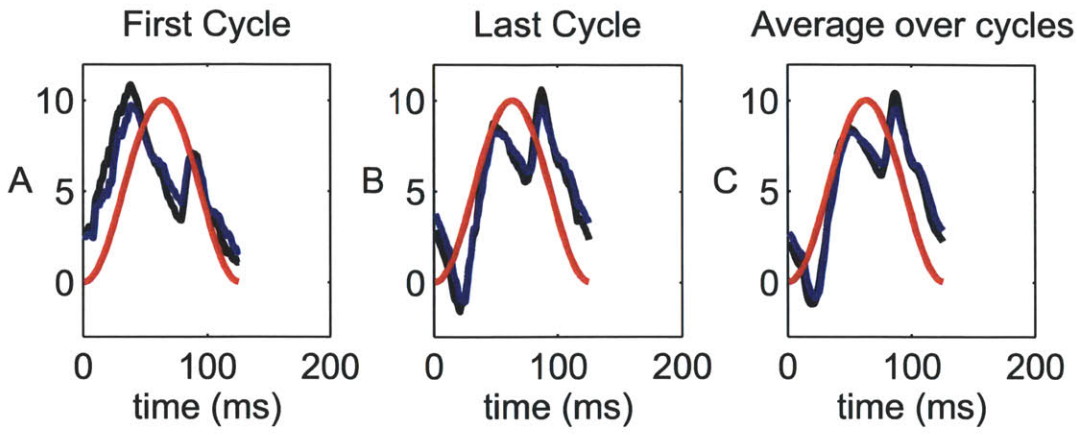

Figure 5-19. Decoded low-velocity stimulus during first and last cycles, and averaged across cycles. (A) First cycle, (B) Last cycle. The figure seems to indicate that, in each cycle, the low-velocity stimulus comprises of two deflections. This would explain the two distinct peaks in the correlation plot for the low-velocity stimulus (Fig. 5-9).

The model shows that the stimulus modulates each of the non-simultaneous and simultaneous events, at all velocities (Fig. 5-12*A*). We measure changes in stimulus-induced modulation of thalamic firing synchrony as changes in the contribution of the stimulus to the instantaneous rate of the simultaneous-spiking ('11') event at the one ms time-scale. Across the population, the model shows strong changes in zero-lag stimulus-induced thalamic firing synchrony at high and medium velocities, which are stronger than the stimulus' modulation of the non-simultaneous events at those velocities (Figs. 5-12*A*). We also found that the stimulus modulation of the simultaneous event is similar for high and medium-velocity stimuli, and an order of magnitude stronger than for the low-velocity stimulus (Fig. 5-14*A*). Across the population, there was no evidence of zero-lag stimulus-induced thalamic firing synchrony for the low-velocity stimulus (Fig. 5-14*A*). These changes/features in/of zero-lag thalamic firing synchrony were also observed when neurons' intrinsic dynamics were taken into account using the correlation $\rho[i]$ (Figs. 5-9*A*, 5-10*A*, 5-15*A*, 5-16*A*), thus confirming previous findings [42]. We'd like to emphasize the fact that the observed changes in thalamic firing synchrony mirror rapid changes in whisker deflection. Indeed, we found that the maximum stimulus modulation of the simultaneous event occurs earlier with respect to the stimulus onset for high and medium-velocity deflections (Fig. 5-13).

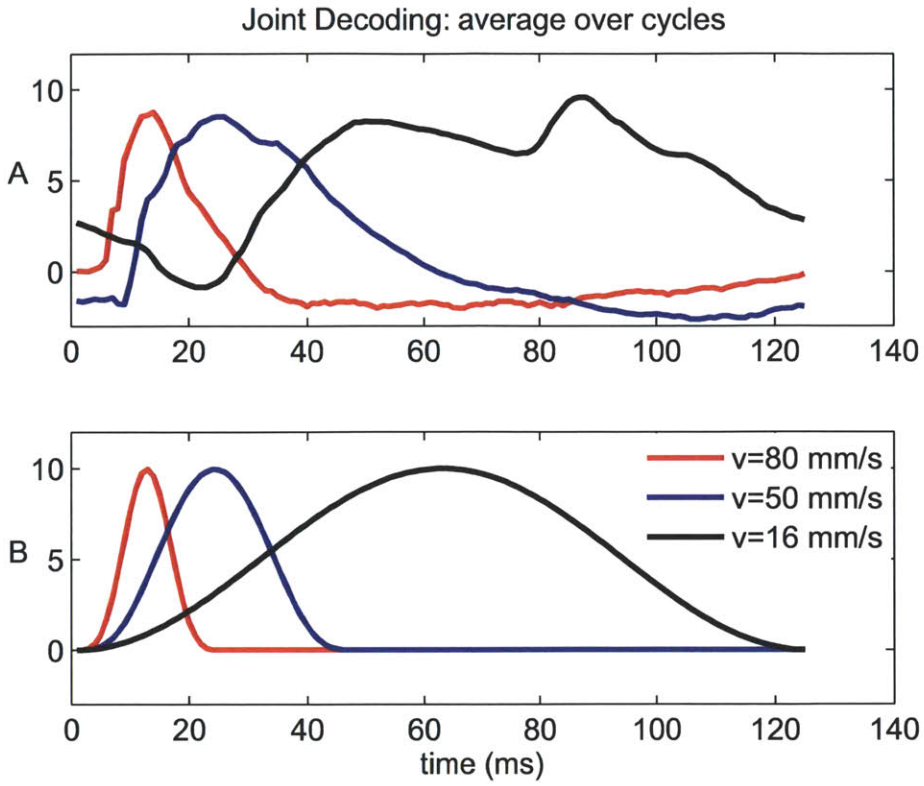

Figure 5-20. Comparison, for each stimulus, of administered stimulus to jointly-decoded stimulus using real data. (A) Average jointly-decoded stimuli over 16 cycles. (B) Administered Stimuli. The figure shows that the decoding algorithm is able to capture the differences between the three stimuli.

The dynamic-inference algorithms, applied to decoding of the low-velocity stimulus, indicate that each cycle of this stimulus may comprise of two successive deflections. Decoding of the low-velocity stimulus using simulated data indicated that the presence of these two deflections is not an artifact of the decoding algorithm. We hypothesize that the secondary deflection may be due to movements of the whisker during the experiment, which it appears are more pronounced at low velocity. Yet another possibility is that the decoding of the secondary deflection is due to inaccuracies in our encoding model. Indeed, even if our model was correct, the assumption that the ideal stimulus is exactly delivered to the whisker does not hold. A model which captures the noise in the stimulus may be more appropriate.

Overall, the results suggest that *individual* pairs of thalamic neurons may employ rapid changes in the instantaneous rate of the simultaneous-spiking event to encode

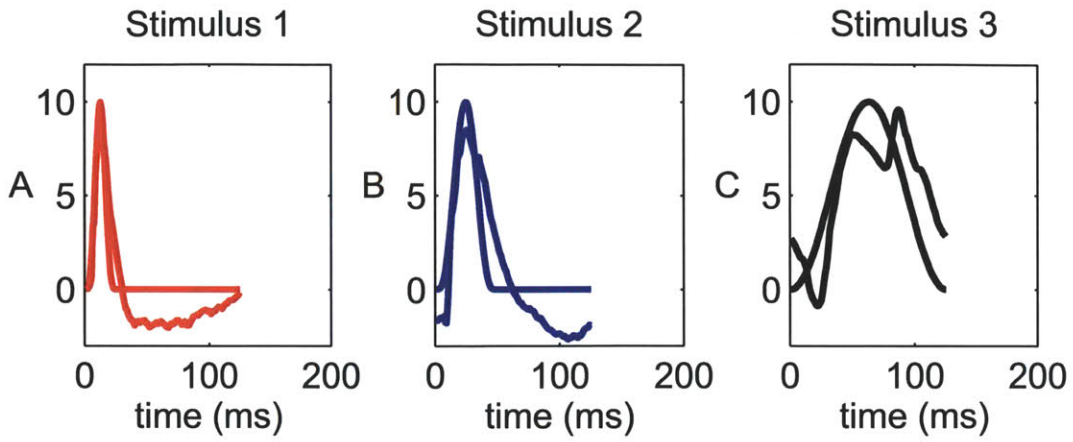

Figure 5-21. Comparison, across stimuli, of administered stimulus to jointly-decoded stimulus using real data. (A) High velocity, (B) Medium velocity, (C) Low-velocity. The decoded stimuli resemble the administered ones. At medium and high velocities, there is a discrepancy between the decoded and administered stimuli in the regions where the administered stimuli are non-zero. This can be attributed to our stochastic continuity constraint which does not allow sharp discontinuities.

whisker movements of varying velocity.

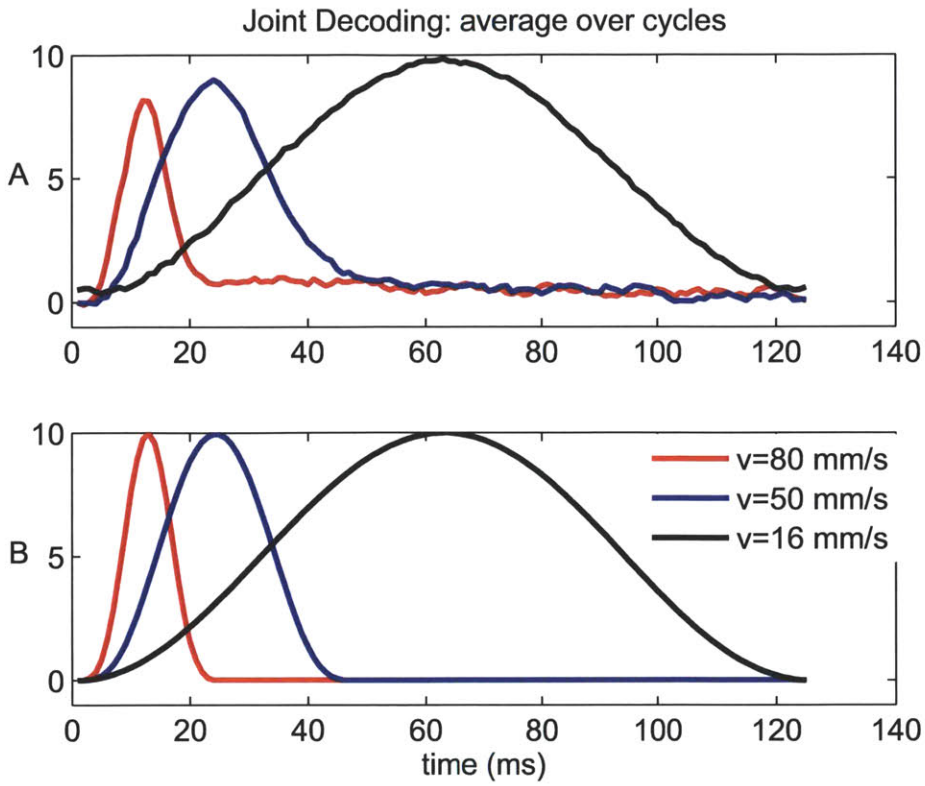

Figure 5-22. Comparison, across stimuli, of administered stimulus to jointly-decoded stimulus using simulated data. (A) Average jointly-decoded stimuli over 16 cycles. (B) Administered Stimuli. The figure shows that the decoding algorithm is able to capture the differences between the three stimuli. The peak values of the high and medium-velocity waveforms are slightly underestimated. This could be due to inaccuracies in our implementation of the simulation algorithm

.

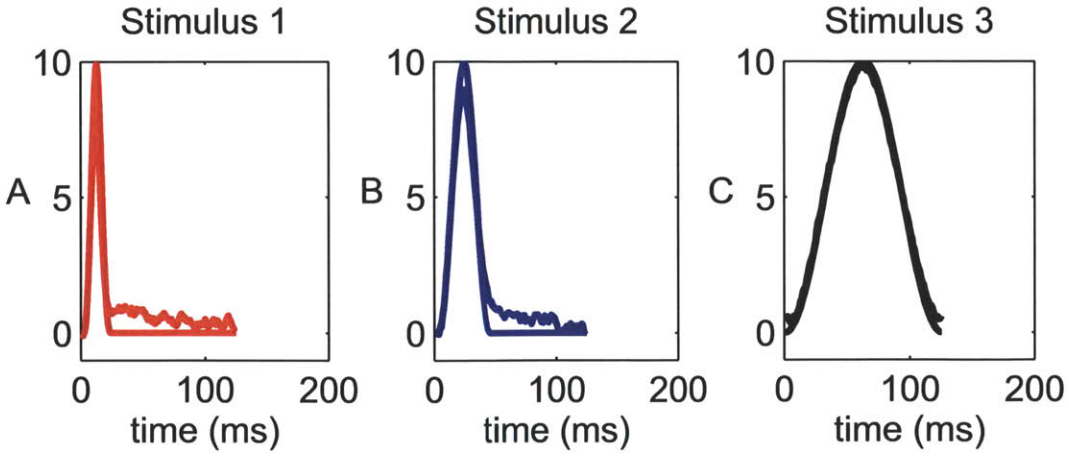

Figure 5-23. Comparison, for each stimulus, of administered stimulus to jointly-decoded stimulus using simulated data. (A) High velocity, (B) Medium velocity, (C) Low-velocity. At medium and high velocities, the peak values of the waveforms are slightly underestimated. This could be due to inaccuracies in our simulation algorithm

Chapter 6

Conclusion

In this chapter, we summarize the contributions of this thesis and point to directions that could be explored further.

6.1 Concluding remarks

In this thesis, we introduce a quite general framework under which one could perform inference based on observations from the class of C-variate point processes with up to $2^C - 1$ degrees of freedom (in a small enough interval), which we termed simultaneous-event multivariate point processes (SEMPPs). We propose a mapping of an SEMPP into a multivariate point-process with no simultaneities, resulting in the so-called disjoint representation of SEMPP. We also introduced a marked point process representation of SEMPP, which gives new efficient algorithms for simulating an SEMPP stochastic process. Starting from a discrete-time approximation to the likelihood of the disjoint representation of SEMPP, we derive the likelihood of the limiting continuous time process and show that it factors into the product of uni-variate point process likelihoods. We also express this continuous time likelihood in terms of the marked point-process representation.

The Jacod likelihood [22] (no simultaneous occurrences) and the likelihood of a uni-variate point process [43] are special cases of the one we derive here. The treatment in [41] considered a similar problem. However, it does not make explicit the relationship to marked point processes with finite mark space, nor does it propose a comprehensive framework for inference.

In practice, model fitting is performed in discrete-time. For static inference, we

propose a parametrization of the discrete-time likelihood of SEMPP which turns it into a multivariate generalized linear model with multinomial observations and logit link [16]. Under certain assumptions, the multinomial GLM becomes equivalent to multiple uni-variate GLMs with Poisson observations and log link. Estimation of the model parameters is performed by maximum likelihood [16]. Under a generalized linear model, the discrete-time likelihood is concave. Therefore, there exists a unique maximum, which can be found using Newton's method. We argue that the use of linear conjugate gradient, to solve the linear system involved at each Newton step, can significantly speed up computations [26]. We demonstrate the possible improvements using data from multiple neuroscience experiments. We provide a set of fast routines for fitting of GLMs of point-process data. These routines are written in Matlab, thus making them accessible to a wide range of researchers. For dynamic inference, we introduce generalized point-process adaptive filters which use the exact and approximate discrete-time likelihoods of the disjoint representation of SEMPP. If one uses the Jacod likelihood instead, we recover the adaptive filters derived in [15].

Arguably, the time-rescaling theorem is the most important result in point-process theory. We suggest a Kolmogorov-Smirnov test to assess the level of agreement between a fitted model and the data, based on the time-rescaling theorem for multivariate point processes with no simultaneities. The test relies on the fact that the disjoint representation of SEMPP is a multivariate point process with no simultaneities, albeit in a higher-dimensional space. Hence, one can readily apply results on rescaling multivariate point processes (with no simultaneities) to marked point processes with finite mark space. The key difference between the said test and that for uni-variate point processes ([9]) is that points with difference marks are rescaled with different conditional intensity functions.

We demonstrate the efficacy of the proposed framework on an analysis of simultaneous recordings from pairs of neurons in the rat thalamus. Our analysis is able to provide a direct estimate of the propensity of pairs of thalamic neurons to fire simultaneously, and the extent to which whisker stimulation modulates this propensity. The results show a strong effect of whisker stimulation on the propensity of pairs of thala-

mic neurons to fire simultaneous, especially for high and medium velocity stimulation. Surprisingly, for a number of pairs, the effect of the stimulus on the simultaneous-spiking event is stronger than its effect on either of the non-simultaneous-spiking events. We also show an application of the dynamic-inference algorithms to decoding of whisker velocity. The decoding example suggests that, at low-velocity, the whisker movement in each cycle comprises of two successive deflections.

6.2 Outlook

6.2.1 Modeling stimulus noise

In modeling the data from pairs of thalamic neurons, we assumed the absence of errors/noise in going from the stimuli to the movement of the whisker. We used the ideal stimuli as inputs to our mGLM fits (Eq. 5.1). The errors in the stimuli could be due to imperfections in the placement of the whisker during the administration of the stimulus. Figure 5-21 shows that the decoded stimuli resemble the administered, ideal stimuli, especially at high and medium velocity. However, there are discrepancies, notably at low velocity. The presence of the secondary deflection is particularly puzzling.

It would be interesting to compare our noiseless model of Eq. 5.1 to one with a random noise component. We would treat that noise as a latent variable with a prior. The inference problem would need to estimate the parameters of the latent variables as well as the fixed parameters of the model, using EM for instance.

6.2.2 Dimensionality reduction

While we set out to solve the problem of dealing multivariate point processes with simultaneities, we do not claim to have solved it in the most elegant of fashion.

A C-variate SEMPP possesses up to $2^C - 1$ degrees of freedom, that is to say, the dimensionality of the ΔN^* process grows exponential with the number of components of the ΔN process. For C small, this would be reasonable. However, as C increases, the problem clearly becomes unmanageable. This points to the necessity of some

dimensionality reduction technique in order for the case of large C to be manageable. It is reasonable to assume that not all $2^C - 1$ degrees of freedom with be 'active' at any given time. The question now becomes: how does one decide which degrees of freedom dominate the probability mass at any given time? By no means is this question posed formally. In fact, if we knew how to pose the problem formally, we would have had a shot at a solution. The main idea here is that the dimensionality of the problem blows up quickly, how does one deal with this in a *principled*, *non*-heuristic fashion.

6.2.3 Large-scale decoding examples using simultaneous events

We demonstrated the techniques developed in this thesis on a data set consisting of simultaneous recordings from *pairs* of neurons in the rat thalamus. Various authors have consider the decoding problem using multivariate point-process data with (conditionally) independent components or no simultaneity. Typically, these decoding problems consist of a large number of neurons that may or may not have been recorded simultaneously. It would be interesting to study the improvements of the SEMPP model for decoding of a stimulus based on a large number of simultaneously-recorded neurons (e.g. place cell data).

6.2.4 Adaptive filtering for the exponential family

The Kalman-like properties of the SEMPP adaptive filters we derive in Chapter 4 are really a property of the exponential family. When we say 'Kalman-like', we are referring to the innovation and gain components of the update equation for the posterior mean. Indeed, one of the key steps in the derivation of the SEMPP adaptive filters is the use of the differential equalities satisfied by the mean and variance of observations from the exponential family. Indeed, if one follows the steps outline in the derivation of the SEMPP adaptive filters, replacing the SEMPP likelihood with that of *any* observations from the exponential family, one can essentially derive a very broad class of filters. These are approximate filters, as the posterior density estimation problem cannot usually be solved in closed form (except in the Gaussian case). Therefore, it is not unreasonable to ask the following question: how good are

the approximations? It would be useful if one could obtain bounds on the extent to which the approximate posterior density differs from the exact one.

Also, from a practical standpoint, are there applications out there that could benefit from these exponential-family adaptive filters?

Appendix A

Chapter 1 Derivations

A.1 Derivation of the Ground Intensity and the Mark pmf

We need to specify (a) the intensity of the ground process (Eq. 2.7) and (b) the distribution of the marks (Eq. 2.8). By definition,

$$\lambda_g^*(t|H_t) = \lim_{\Delta \to 0} \frac{P[\Delta N_{g,t} = 1|H_t]}{\Delta}. \tag{A.1}$$

$$\begin{aligned} P[\Delta N_{g,t} = 1|H_t] &= P\Big[\bigcup_{m=1}^{M-1} \Delta N_{m,t}^* = 1|H_t\Big] && \text{(A.2)} \\ &= \sum_{m=1}^{M-1} P\big[\Delta N_{m,t}^* = 1|H_t\big] + o(\Delta) && \text{(A.3)} \\ &= \sum_{m=1}^{M-1} \lambda_m^*(t|H_t)\Delta + o(\Delta), && \text{(A.4)} \end{aligned}$$

where the second equality follows from the fact that the events $\{\Delta N_{m,t}^* = 1 \cap \Delta N_{k,t}^* = 1\} = \emptyset$ for all (m, k) given full history (i.e. ΔN_t^* has no simultaneities). From here, it is not hard to see that

$$\lambda_g^*(t|H_t) = \sum_{m=1}^{M-1} \lambda_m^*(t|H_t). \tag{A.5}$$

The mark PMF requires a little more work. We are seeking an expression for

$P\left[dN_m^*(t) = 1 | dN_g(t) = 1, H_t\right]$ in terms of the $\lambda_m^*(t|H_t)$'s.

$$P\left[dN_m^*(t) = 1 | dN_g(t) = 1, H_t\right] = \lim_{\Delta \to 0} P\left[\Delta N_{m,t}^* = 1 | \Delta N_{g,t} = 1, H_t\right] \tag{A.6}$$

$$= \lim_{\Delta \to 0} \frac{P\left[\Delta N_{m,t}^* = 1 | H_t\right]}{P\left[\Delta N_{g,t} = 1 | H_t\right]} \tag{A.7}$$

$$= \lim_{\Delta \to 0} \frac{\lambda_m^*(t|H_t)\Delta + o(\Delta)}{\lambda_g^*(t|H_t)\Delta + o(\Delta)} \tag{A.8}$$

$$= \frac{\lambda_m^*(t|H_t)}{\lambda_g^*(t|H_t)}, \tag{A.9}$$

$m = 1, \cdots, M-1$, so that the marks follow a multinomial distribution with probabilities given as above.

A.2 Expressing the Discrete-time Likelihood of Eq. 2.12 in Terms of a Discrete Form of the MkPP Representation

$$P\left[\Delta N^*\right] = \prod_{i=1}^{I} \prod_{m=1}^{M-1} \frac{\left(\lambda_m^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*}}{\left(\lambda_g^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*}} \left(\lambda_g^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}} + o(\Delta^L) \tag{A.10}$$

$$= \prod_{i=1}^{I} \prod_{m=1}^{M-1} \left(\frac{\lambda_m^*[i|H_i]\Delta}{\lambda_g^*[i|H_i]\Delta}\right)^{\Delta N_{m,i}^*} \left(\lambda_g^*[i|H_i]\Delta\right)^{\Delta N_{m,i}^*} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}} + o(\Delta^L) \tag{A.11}$$

$$= \prod_{i=1}^{I} \prod_{m=1}^{M-1} \left(\frac{\lambda_m^*[i|H_i]\Delta}{\lambda_g^*[i|H_i]\Delta}\right)^{\Delta N_{m,i}^*} \left(\lambda_g^*[i|H_i]\Delta\right)^{\Delta N_{g,i}^*} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}} + o(\Delta^L) \tag{A.12}$$

$$= \prod_{i=1}^{I} \prod_{m=1}^{M-1} \left(\frac{\lambda_m^*[i|H_i]\Delta}{\lambda_g^*[i|H_i]\Delta}\right)^{\Delta N_{m,i}^*} \prod_{i=1}^{I} \left(\lambda_g^*[i|H_i]\Delta\right)^{\Delta N_{g,i}} \left(1 - \lambda_g^*[i|H_i]\Delta\right)^{1-\Delta N_{g,i}} + o(\Delta^L). \tag{A.13}$$

Appendix B

Gradient vector and Hessian matrix of multinomial GLM log-likelihood

We derive the gradient vector and the Hessian matrix of a GLM with multinomial observations and logit link. We do this for a single observation/covariate pair and easily generalize it to the case of multiple observations.

We observe the data in the form of $(\Delta N_i^*, x_i)$ pairs, where ΔN_i^* is an $M-1$-length vector corresponding to one of M possibles multinomial outcomes and x_i is a d-length vector of covariates/features associated with ΔN_i^*. The log likelihood of a single $(\Delta N_i^*, x_i)$ pair is given by:

$$L\big(\Delta N_i^*, x_i; \beta\big) = \sum_{m=1}^{M-1} \Delta N_{m,i}^* \beta_m' x_i - \log\left(1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\}\right).$$

Let $g_i^{(m)}(\beta)$ bet the partial derivative of this log likelihood with respect to β_m:

$$\begin{aligned} g_i^{(m)}(\beta) &= \frac{\partial L\big(\Delta N_i^*, x_i; \beta\big)}{\partial \beta_m} \\ &= \left(\Delta N_{m,i}^* - \frac{\exp\{\beta_m' x_i\}}{1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\}}\right) x_i, \\ &= \big(\Delta N_{m,i}^* - \lambda_m^*[i|H_i, \beta]\Delta\big)\, x_i. \end{aligned} \tag{B.1}$$

Therefore,

$$\begin{aligned} g_i(\beta) &= \frac{\partial L\big(\Delta N_i^*, x_i; \beta\big)}{\partial \beta} \\ &= (\Delta N_i^* - \lambda^*[i|H_i, \beta]\Delta) \otimes x_i \\ &= X_i' \left(\Delta N_i^* - \lambda^*[i|H_i, \beta]\Delta\right), \end{aligned}$$

where $A \otimes B$ is the Kronecker product of matrices A and B, X_i is an $(M-1) \times (M-1)d$ block-diagonal matrix with x_i' repeated $M-1$ times along the diagonal, and $\lambda^*[i|H_i, \beta] = (\lambda_1^*[i|H_i, \beta], \cdots, \lambda_{M-1}^*[i|H_i, \beta])'$.

Let $H_i^{(m,m')}(\beta)$ be the partial derivative of $g_i^{(m)}(\beta)$ with respect to $\beta_{m'}$:

$$\begin{aligned} H_i^{(m,m')}(\beta) &= \frac{\partial g_i^{(m)}(\beta)}{\partial \beta_{m'}} \\ &= -\frac{\delta_{m,m'} \cdot \left(\exp\{\beta_m' x_i\}(1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\})\right) - \exp\{\beta_m' x_i\} \cdot \exp\{\beta_{m'}' x_i\}}{\left(1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\}\right)^2} \cdot x_i x_i' \\ &= -\left(\delta_{m,m'} \frac{\exp\{\beta_m' x_i\}}{1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\}} - \frac{\exp\{\beta_m' x_i\} \cdot \exp\{\beta_{m'}' x_i\}}{\left(1 + \sum_{m=1}^{M-1} \exp\{\beta_m' x_i\}\right)^2}\right) \cdot x_i x_i' \\ &= -\left(\delta_{m,m'} \cdot \lambda_m^*[i|H_i, \beta]\Delta - \lambda_m^*[i|H_i, \beta]\lambda_{m'}^*[i|H_i, \beta]\Delta^2\right) x_i x_i'. \end{aligned} \tag{B.2}$$

Therefore,

$$\begin{aligned} H_i(\beta) &= \frac{\partial^2 L\big(\Delta N_i^*, x_i; \beta\big)}{\partial^2 \beta} \\ &= -\left(\text{diag } \lambda^*[i|H_i, \beta]\Delta - \lambda^*[i|H_i, \beta]\lambda^*[i|H_i, \beta]'\Delta^2\right) \otimes x_i x_i' \\ &= -W_i(\beta) \otimes x_i x_i' \\ &= -X_i' W_i(\beta) X_i, \end{aligned}$$

where diag $\lambda^*[i|H_i, \beta]$ is an $(M-1) \times (M-1)$ diagonal matrix whose diagonal entries correspond to the elements of the vector $\lambda^*[i|H_i, \beta]$.

Finally, the gradient vector $g(\beta)$ and the Hessian matrix $H(\beta)$ for all I observations

are given by

$$g(\beta) = \sum_{i=1}^{I} g_i(\beta), \text{ and}$$

$$H(\beta) = \sum_{i=1}^{I} H_i(\beta).$$

Note that these can also be expressed in matrix form as follows:

$$g(\beta) = X' \left(\Delta N^* - \lambda^*[\beta]\Delta \right), \text{ and}$$

$$H(\beta) = -X'W(\beta)X,$$

where X is an $(M-1)I \times (M-1)d$ matrix with the X_i's stacked on top of each other, W is an $(M-1)I \times (M-1)I$ block-diagonal matrix with the W_i's on the diagonal, and ΔN^* as well as $\lambda^*[\beta]$ are $(M-1)I$-length column vectors of the ΔN_i^*'s and $\lambda^*[i|H_i, \beta]$'s stacked on top of each other.

Gradient vector and Hessian matrix of approximate likelihood: We saw previously that, for small Δ, the GLM for the joint process is approximately equivalent to $M-1$ independent uni-variate GLMs with Bernoulli observations and log link. The gradient vector and Hessian matrix using this approximation are straightforward to obtain from those of a uni-variate GLM with Bernoulli observations and log link [16]. Therefore, we only specify the gradient vector and the Hessian matrix for a single observation and one of $M-1$ uni-variate GLMs. The important thing to realize here is that, using this approximation, the Hessian is block-diagonal, with each block corresponding to one of the $M-1$ uni-variate GLMs. Assuming that the discrete-time likelihood can be approximated as in Equation 2.18 and that $\log \lambda_m^*[i|H_i]\Delta = \beta_m' x_i$,

$$\begin{aligned} g_i^{(m)}(\beta) &\approx \left(\Delta N_{m,i}^* - \lambda_m^*[i|H_i]\Delta\right) x_i = \left(\Delta N_{m,i}^* - \exp\{\beta_m' x_i\}\right) x_i, \\ H_i^{(m,m')}(\beta) &\approx -\delta_{m,m'} \lambda_m^*[i|H_i]\Delta \cdot x_i x_i' = -\delta_{m,m'} \exp\{\beta_m' x_i\} \cdot x_i x_i'. \end{aligned}$$

One may also think of the above equations as obtained from Equations B.1 and B.2 by dropping the terms involving Δ^2, which we assume are $o(\Delta)$.

Appendix C

Second-order statistics of a multinomially-distributed random vector

Consider an M-sided die with sides labeled $0, \cdots, M-1$. The said die is thrown R times and let the outcome of the r^{th} trial be an $(M-1)$-length indicator vector $y^{(r)}$ whose m^{th} entry $y_m^{(r)}$ is 1 if we observed side m ($m \in \{1, \cdots, M-1\}$). Note that $y^{(r)} = (0, 0, \cdots, 0)'$ corresponds to outcome 0 being observed at the r^{th} trial. Let $(\pi_1, \cdots, \pi_{M-1})'$ be an $M-1$-length vector of probabilities for sides 1 to $M-1$. In this experiment, we are interested in the joint pmf of the $(M-1)$-length random vector $y = \sum_{r=1}^{R} y^{(r)}$, whose m^{th} entry y_m indicates the number of times we observed side m. For instance, in the case of R i.i.d. Bernoulli trials ($M = 2$), $y \in \{0, 1, ..., R\}$ is scalar-valued and follows a binomial distribution with probability of success π_1. The multinomial distribution is the natural generalization of the binomial to the case when M is arbitrary but finite. Indeed, the distribution of the random vector y is given by:

$$P\left[y = (y_1, y_2, \cdots, y_{M-1})'\right] = \frac{R!}{y_1! y_2! \cdots y_{M-1}! (R - y_1 - y_2 - \cdots - y_{M-1})!} \pi_1^{y_1} \pi_2^{y_2} \quad \text{(C.1)}$$

$$\cdots \pi_{M-1}^{y_{M-1}} \left(1 - \pi_1 + \pi_2 + \cdots + \pi_{M-1}\right)^{R - y_1 - y_2 - \cdots - y_{M-1}}, \quad \text{(C.2)}$$

We note the following properties of the multinomial distribution which will be helpful in deriving its second-order statistics:

1. Each y_m follows a binomial distribution with probability of success π_m.

2. Every $(y_m, y_{m'})$ pair, $m \neq m'$ follows a tri-nomial ($M = 3$) distribution with R trials and probability vector $(\pi_m, \pi_{m'})$.

3. Consider the trinomially distributed pair $(y_m, y_{m'})$ mentioned above. Conditioned on y_m, $y_{m'}$ follows a binomial distribution with $R - y_m$ trials and probability of success $\frac{\pi_{m'}}{1-\pi_m}$.

Without loss of generality, let us compute the mean and covariance of the pair (y_1, y_2). The means are easily obtained by using the fact that y_1 and y_2 both have binomial marginals: $E[y_1] = R \cdot \pi_1$ and $E[y_2] = R \cdot \pi_2$. The covariance of y_1 and y_2 requires a little more effort:

$$E\left[(y_1 - E[y_1])(y_2 - E[y_2])\right] = E[y_1 y_2] - E[y_1]E[y_2]. \tag{C.3}$$

As we have already obtained the means, we focus on the 1st term in the right-hand side of the equality above:

$$\begin{aligned}
E[y_1 y_2] &= E_{y_1}\left[E_{y_2|y_1}[y_1 y_2|y_1]\right] && \text{(C.4)}\\
&= E_{y_1}[y_1 \underbrace{E_{y_2|y_1}[y_2|y_1]}_{(R-y_1)\frac{\pi_2}{1-\pi_1}}] && \text{(C.5)}\\
&= \frac{\pi_2}{1-\pi_1}\left(RE[y_1] - E[y_1^2]\right) && \text{(C.6)}\\
&= \frac{\pi_2}{1-\pi_1}\left(R^2\pi_1 - (R\pi_1(1-\pi_1) + R^2\pi_1^2)\right) && \text{(C.7)}\\
&= \frac{\pi_2}{1-\pi_1}\left(R^2\pi_1(1-\pi_1) - R\pi_1(1-\pi_1)\right) && \text{(C.8)}\\
&= R^2\pi_1\pi_2 - R\pi_1\pi_2, && \text{(C.9)}
\end{aligned}$$

where the 2nd equality uses the fact that conditioned on y_1, y_2 follows a binomial distribution with $R - y_1$ trials and probability of success $\frac{\pi_{l'}}{1-\pi_l}$. The 4th equality results from expressing the 2nd moment of y_1 in terms of its mean and variance. The

remaining equalities follow from trival algebraic manipulations. Therefore,

$$E\left[(y_m - E[y_m])(y_{m'} - E[y_{m'}])\right] = \underbrace{R^2\pi_m\pi_{m'} - R\pi_m\pi_{m'}}_{E[y_m y_{m'}]} - \underbrace{(R\pi_m)(R\pi_{m'})}_{E[y_m]E[y_{m'}]} \quad \text{(C.10)}$$

$$= -R\pi_m\pi_{m'},\ m \neq m'. \quad \text{(C.11)}$$

Bibliography

[1] M. Abeles and G. L. Gerstein. Detecting spatiotemporal firing patterns among simultaneously recorded single neurons. *J. Neurophysiol.*, 60:909–924, 1988.

[2] J.M. Alonso, W.M. Usrey, and R.C. Reid. Precisely correlated firing in cells of the lateral geniculate nucleus. *Nature*, 383(6603):815–819, 1996.

[3] S. Amari. Information geometry on hierarchy of probability distributions. *IEEE Trans. Inform. Theory*, IT-47(5):721–726, July 2001.

[4] D. Böhning. Multinomial logistic regression algorithm. *Annals of the Inst. of Statistical Math.*, 44:197–200, 1992.

[5] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis, Forecasting and Control.* Prentice-Hall, Englewood Cliffs, NJ, 3 edition, 1994.

[6] D. R. Brillinger. Nerve cell spike train data analysis. *J. Amer. Stat. Assoc.*, 87:260–271, 1992.

[7] D. R. Brillinger, H. L. Bryant, and J. P. Segundo. Identification of synaptic interactions. *Biol. Cybern.*, 22:213–220, 1976.

[8] C. D. Brody. Correlations without synchrony. *Neural Computation*, 11:1537–1551, 1999.

[9] E. N. Brown, R. Barbierri, V. Ventura, R. Kass, and L. Frank. The time-rescaling theorem and its application to neural spike train data analysis. *Neural Computation*, 14:325–346, 2002.

[10] E.N. Brown, L.M. Frank, D. Tang, M.C. Quirk, and M.A. Wilson. A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–7425, 1998.

[11] T. Brown and M. Nair. A simple proof of the multivariate random time change theorem for point processes. *J. Appl. Probab.*, 25:210–214, 1988.

[12] R.M. Bruno and B. Sakmann. Cortex is driven by weak but synchronously active thalamocortical synapses. *Science*, 312(5780):1622, 2006.

[13] E. S. Chornoboy, L. P. Schramm, and A. F. Karr. Maximum likelihood identification of neural point process systems. *Biol. Cybern.*, 59:265–275, 1988.

[14] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*, volume 1. Springer, 2nd edition, 2003.

[15] U. T. Eden, L. M. Frank, V. Solo, and E. N. Brown. Dynamic analyis of neural encoding by point process adaptive filtering. *Neural Computation Letters*, 16:971–998, 2004.

[16] L. Fahrmeir and G. Tutz. *Multivariate Statistical Modeling Based on Generalized Linear Models.* Springer, 2nd edition, 2001.

[17] G. L. Gerstein and D. H. Perkel. Simultaneously recorded trains of action potentials: analysis and functional interpretation. *Science*, 164:828–830, 1969.

[18] S. Grün, M. Diesmann, and A. Aersten. Unitary events in multiple single-neuron spiking activity: II. nonstationary data. *Neural Computation*, 14:81–119, 2002.

[19] R. Gütig, A. Aersten, and S. Rotter. Statistical significance of coincident spikes: count-based versus rate-based statistics. *Neural Computation*, 14:121–153, 2002.

[20] J.A. Hartings and D.J. Simons. Thalamic relay of afferent responses to 1-to 12-Hz whisker stimulation in the rat. *J. Neurophysiol.*, 80(2):1016, 1998.

[21] R. Haslinger, G. Pipa, B. Lima, W. Singer, E. N. Brown, and S. S. Neuenschwander. Beyond the receptive field: predicting v1 spiking during natural scenes vision. *In preparation*, 2010.

[22] J. Jacod. Multivariate point processes: predictable projection, radon-nikodym derivatives, representation of martingales. *Probability Theory and Related Fields*, 31(3):235–253, 1975.

[23] A. Johnson and S. Kotz. *Distributions in statistics: Continuous univariate distributions*, volume 2. Wiley, New York, 1970.

[24] A. F. Karr. *Point processes and their statistical inference.* Dekker, New York, 1991.

[25] R.E. Kass, R.C. Kelly, and W.L. Loh. Assessment of synchrony in multiple neural spike trains using log linear point process models. *Annals of Applied Statistics*, 2010. to appear.

[26] P. Komarek and A. W. Moor. Making logistic regression a core data mining too with TR-IRLS. In *Proc. 5th IEEE international conference on data mining*, pages 685–688, Houston, USA, 2005.

[27] B. Krishnapuram, A. J. Hartemink, L. Carin, and M. A. T. Figueiredo. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Trans. Pattern Anal. and Mach. Int.*, 27(6):957–968, 2005.

[28] E. M. Maynard, C. T. Nordhausen, and R. A. Normann. The utah intracortical electrode array: a recording structure for potential brain-computer interfaces. *Electroencephalogr. Clin. Neurophysiol.*, 102:228–239, 1997.

[29] P. Meyer. Démonstration simplifiée d'un théorème de knight. *In Séminaire de Probabilités V, Univ. Strasbourg, Lecture Notes in Math.*, 191:191–195, 1971.

[30] T. P. Minka. A comparison of numerical optimizers for logistic regression. `http://research.microsoft.com/ minka/papers/logreg/`, 2003.

[31] H. Nakahara and S. Amari. Information-geometric measure for neural spikes. *Neural Computation*, 14:2269–2316, 2002.

[32] Y. Ogata. On lewis' simulation method for point processes. *IEEE Trans. Inform. Theory*, 27(1):23–31, January 1981.

[33] M. Okatan, M. A. Wilson, and E. N. Brown. Analyzing functional connectivity using a network likelihood model of neural ensemble spiking activity. *Neural Computation*, 17:1927–1961, 2005.

[34] G. Pipa and S. Grün. Non-parametric significance estimation of joint-spike events by shuffling and resampling. *Neuocomputing*, 52–54:31–37, 2002.

[35] E. Plourde, B. Delgutte, and E. N. Brown. A Point Process Model for Auditory Neurons Considering both their Intrinsic Dynamics and the Spectro-Temporal Properties of an Extrinsic Signal. *Submitted to IEEE Transactions on Biomedical Engineering*, 2010.

[36] S.A. Roy and K.D. Alloway. Coincidence detection or temporal integration? What the neurons in somatosensory cortex are doing. *Journal of Neuroscience*, 21(7):2462, 2001.

[37] S. V. Sarma, U. T. Eden, M. L. Cheng, Z. M. Williams, R. Hu, E. Eskandar, and E. N. Brown. Using Point Process Models to Compare Neural Spiking Activity in the Subthalamic Nucleus of Parkinson's Patients and a Healthy Primate. *IEEE Transactions on Biomedical Engineering*, 57(6):1297–1305, 2010.

[38] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical Report CS-94-125, Carnegie Mellon University, 1994.

[39] H. Shimazaki, S. Amari, E. N. Brown, and S. Grün. State-space analysis of time-varying correlations in parallel spike sequences. In *icassp*, pages 3501–3504, Taipei, Taiwan, April 2009.

[40] DJ Simons and GE Carvell. Thalamocortical response transformation in the rat vibrissa/barrel system. *J. Neurophysiol.*, 61(2):311, 1989.

[41] V. Solo. Likelihood functions for multivariate point processes with coincidences. In *Proc. IEEE Conf. Dec. & Contr.*, volume 3, pages 4245–4250, December 2007.

[42] S. Temereanca, E.N. Brown, and D.J. Simons. Rapid changes in thalamic firing synchrony during repetitive whisker stimulation. *Journal of Neuroscience*, 28(44):11153–11164, 2008.

[43] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue, and E. N. Brown. A point-process framework for relating neural spiking activity to spiking history, neural ensemble, and extrensic covariate effects. *J. Neurophysiol.*, 93:1074–1089, 2005.

[44] W.M. Usrey, J.M. Alonso, and R.C. Reid. Synaptic interactions between thalamic inputs to simple cells in cat visual cortex. *Journal of Neuroscience*, 20(14):5461, 2000.

[45] V. Ventura, C. Cai, and R. E. Kass. Statistical assessment of time-varying dependency between two neurons. *J. Neurophysiol.*, 94:2940–2947, 2005.

[46] D. Vere-Jones and F. Schoenberg. Rescaling marked point processes. *Australian and New Zealand Journal of Statistics*, 46(1):133–143, 2004.

[47] M. A. Wilson and B. L. McNaughton. Dynamics of the hippocampal ensemble code for space. *Science*, 261:1055–1058, 1993.