Dynamical implications of network statistics

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Patrick Campbell

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Nykamp, Duane

August, 2014

Acknowledgements

First and foremost I would like to express my deepest gratitude to my advisor Duane Nykamp for his guidance and generosity of time and mind. Also, thanks to Michael Buice for sharing his knowledge of path integral methods over the course of the last two years. Many thanks to my committee, Hans Othmer, Jasmine Foo, and Tay Netoff. With respect to the research presented here, I owe thanks to Liqiong Zhao, Chris Kim, Yu Hu, and Badal Joshi for their stimulating conversations. Thanks to Eric Shea-Brown, and his group for making me feel very welcome at University of Washington last year, and sharing their positivity and encouragement. Thanks to Anne Springfield and Aaron Berdofe for reading through my dissertation and for fixing my commas.

Graduate school would not have been half as educational or formative without many conversations and seminars with my peers. In particular, Jonas Karlsson and Trevor Bain have played critical roles in my thinking about nearly everything from general relativity to artificial intelligence. I will be forever grateful for their influences.

My family has always given me tremendous support. I thank you all, especially my wife, Sandhya. Without her patience and partnership I could never have done this.

Finally, a very special thanks to Kevin Vixie for being my mentor and inspiring me to pursue mathematics.

Dedication

For Sandhya, my love.

Abstract

Dynamics on large networks can be highly complex. I present several methods for investigating the effects of network structure/statistics on rate dynamics and spike correlations. The dynamical models under consideration come from computational neuroscience, but these methods may generalize to other contexts. The thesis focuses on both network constructions and dynamics on networks.

I present two approaches to network constructions: random networks, and networks with patterns. For random networks, I give a generalization of the expected degree model (EDM) and a formulation of the EDM and its generalization which is invariant of the number of nodes. This generalization allows one to produce random networks which have nontrivial second and third order correlations among edges. I also introduce a method for constructing networks with nontrivial structures/patterns at multiple scales. I investigate the spectral properties of the resulting networks and extend the method to include stochastic elements.

The singular value decomposition (SVD) is a powerful tool with many applications. I review its application to network adjacency matrices, including a known result which relates the singular values of an adjacency matrix to a measure of its randomness. Further, I demonstrate that for several random network models the degree sequence is the most significant feature of the connectivity.

The primary dynamical model I consider is the Poisson spiking model (PSM). I derive first and second order statistics for the PSM using a path integral formalism.

The major contribution of this work is a dimension reduction method for dynamics on a network using the SVD. I demonstrate how one can use these low rank representations of the connectivity, together with the reduced equations to approximately recover node-specific activity. Thus, not only do I present methods that reduce the number of dynamical variables, but I show how the dynamics of the full system may be decomposed into the reduced variables and network structure.

Contents

List of Figures

Chapter 1

Introduction

The human brain is a vast network of neurons. The structure of the brain's network is rich [49]. Brain mapping programs like the human connectome project [51], [50] have made some progress mapping connectivity between brain regions. Also, a number of small circuits [44] are known. Nevertheless, specific connectivities for even moderately sized subnetworks of the brain remain largely unknown. In lieu of specific connectivity maps, there have been a number of studies regarding statistical aspects of brain networks, for example [47], [41]. Not surprisingly, it has been found that neuronal networks exhibit nonrandom characteristics [5]. Along these lines of research, a number of statistical network models have been proposed to generate networks which display statistics similar to those found in brain networks [45], [48], [16].

Investigations into the interplay between network structure and function are numerous, and complement the studies of network structures in the brain. There has been great progress made towards understanding the effects of network statistics or motifs on dynamics. For example, it has been found that increasing the frequency of chain motifs (i connects to j connects to k) increases the tendency of the network to synchronize [57], [58]. Further work considering the effects of network statistics on statistics of activity on networks include [29], [52]. Another example of research along these lines is the article by Netoff, et. al. [37] in which parameters of small-world networks [55] were seen to affect the tendency of excitatory networks to exhibit activity similar to epileptic seizures.

In this thesis, I will present work regarding network models, and dynamics on large networks. We will approach the topic of network structure from two directions: (1) random networks, (2) structured networks. With respect to random/statistical networks, the main point of departure will be the expected degree model of Chung and Lu [11]. The expected degree model admits control over second order edge correlations. I will give a generalization in section 3.3 of the expected degree model which allows one to control third order edge correlations as well. This generalized expected degree model will be used in section 5.3 to investigate the accuracy of certain approximations to adjacency matrices which will be applied in chapter 8. In terms of structured networks, I will introduce an intuitive method for constructing networks with patterns/symmetries which utilizes the Kronecker product for matrices in chapter 4. Finally, the two approaches will be combined to construct hybrid network models which exhibit different characteristics at different scales in section 4.5. The hybrid model will be used in example 8.4.1 to demonstrate a system in which traveling waves in network activity may be attributed to macroscopic symmetry despite the absence of any nontrivial symmetries in the network.

Dynamically, we will work primarily with the Poisson spiking model (PSM) [22]. I will apply path integral techniques [10] in chapter 7 to derive equations for the first and second order statistics for the PSM which will be used in chapter 8. The main contribution of this thesis is the introduction of dimension reduction techniques for dynamics on large networks (chapter 8). The main idea behind the technique is to use the singular value decomposition [31], and low rank approximations of the adjacency matrix in order to close population wide rate equations, and reduce the number of dynamical variables. Moreover, I will demonstrate how one can use these low rank representations of the connectivity, together with the reduced equations to approximately recover node specific activity. Thus, not only will I present methods that reduce the number of dynamical variables, but I'll show how the dynamics of the full system may be decomposed into the reduced variables plus network structure.

The outline of the thesis is as follows. *Novel work* is indicated by italics.

Part I Networks

- **Chapter 2** covers preliminaries regarding indexing conventions and basic definitions of network motif statistics, and provides alternative expressions for the motif statistics.

- **Chapter 3** discusses random network models. The expected degree model [11], and the SONET model [57] are reviewed. Then *a generalization of the expected degree model*, and *a formulation of the expected degree model and its generalization which is invariant of the number of nodes* are presented. The chapter is concluded with *examples which illustrate the techniques*

- **Chapter 4** presents *a method for constructing networks with nontrivial structure*. The *spectral properties of the resulting networks* is discussed, and the method is *extended to include stochastic elements*.

- **Chapter 5** reviews the application of the singular value decomposition **(SVD)** to network adjacency matrices. A known result of Butler's [8] which relates the singular values of an adjacency matrix to a measure of its randomness is reviewed. *Evidence showing that for several random network models the degree sequence is the most significant feature of the connectivity* is presented.

Part II Dynamics

- **Chapter 6** reviews the definition of the Poisson spiking model **(PSM)** which will serve as the primary dynamical model for the remainder of the thesis.

- **Chapter 7** summarizes the basics of the path integral formulation of stochastic differential equations. This framework is then applied to the PSM and estimates for the mean and covariances of the model are derived.

- **Chapter 8** presents a *dimension reduction method, the* low rank reduction*, for dynamics on a network based on the SVD.*

- **Chapter 9** is an addendum. It discusses *ongoing work regarding a Langevin equation formulation of the PSM using path integral methods.*

Part I

Networks

Chapter 2

Background: Observed statistics of networks

There are many ways to characterize networks. They are at least as combinatorially rich as the natural numbers, and there is no single measure which captures the variety of all networks with a large number of nodes. Throughout this thesis I will be concerned with the relationship between network structure/statistics and dynamics. In this chapter, I'll review several statistical measures of networks, which will be referred to frequently in the remainder of the thesis. For a given network, these quantities reflect the prevalence of certain simple motifs. As such, they provide natural measures to compare other quantities against in order to investigate the effects of these simple structures.

I will begin by establishing conventions regarding nomenclature, and indexing. Then I will review the basic definitions of low order network statistics, and provide some alternative expressions which will reveal more clearly the ways that these statistics reflect the overall structure of a network. I'll conclude this chapter with expressions for derivatives of the statistics with respect to edge weights. The second order statistics defined here can be found in [58], while third order statistics can be found in [20].

2.1 Networks, basic definitions and conventions

A *directed graph* consists of a collection of vertices $V = \{1, \ldots, N\}$ and a collection of edges $E \subset V \times V$. I will refer to directed graphs as *networks*, and to the vertices of a network as *nodes*. Throughout, I will assume that networks do not have *loops*, edges to and from the same node, and that there can be at most one edge from a node to another node.

Given a network the associated adjacency matrix W is defined componentwise by

$$W^i_j = 1 \quad \Longleftrightarrow \quad \text{there is an edge from } j \text{ to } i.$$

Note the indexing convention:

$$W^{\text{to}}_{\text{from}}, \quad \text{or} \quad W^{\text{in}}_{\text{out}}$$

Throughout, upper indices will refer to the rows of a matrix while lower indices will refer to columns. Thus the i, j^{th} component of W is W^i_j. A row vector will have indices low, and column vectors will have indices high.

I will generally not make a distinction between a network and its adjacency matrix. In statements, such as *Let W be an adjacency matrix*, a network is implicit. Given an adjacency matrix W, we define the *in* and *out degree sequences/vectors* d^i_{in} and d^{out}_j componentwise by

$$d^i_{\text{in}} = \sum_j W^i_j, \quad \text{and} \quad d^{\text{out}}_j = \sum_i W^i_j.$$

The words *in*, and *out* in this notation may be unnecessary given the upper/lower indexing convention, but I'll include them for clarity. Notice that d_{in} is a column vector, and d^{out} is a row vector. I will have occasion to take the outer product of these vectors and will simply write $d_{\text{in}} d^{\text{out}}$ to denote this outer product.

For an arbitrary matrix A, we define $|A|$ by

$$|A| = \sum_{ij} |A^i_j|.$$

Note that for an adjacency matrix W, $|W|$ is the number of edges in the network.

I will also have occasion to employ the *Einstein summation convention*. If an index appears as a superscript and a subscript in an expression, then it should be understood that the index is summed over the values it can assume, unless explicitly stated otherwise. Thus, for example, we have

$$u_i v^i = \sum_i u_i v^i = v^i u_i.$$

On the other hand, there is no sum in an expression like $u^i v^i$, unless explicitly indicated.

2.2 First and second order observed statistics

By *observed statistics*, I mean global (or population level) quantities that we compute from a given fixed network. For a fixed adjacency matrix W, there are a number of such statistics. For example, one could consider the number of nodes N, or the number of edges $|W|$.

Again, W^i_j refers to the i^{th} row, j^{th} column of W, and in terms of the network $W^i_j = 1$ means there is an edge from node j to node i. We define the *edge probability*, or *edge density* of W by

$$p = \frac{|W|}{N(N-1)} \approx \frac{|W|}{N^2}.$$

The value p indicates how many edges are present relative to the number of possible edges. In general, various products of W, and W^T will correspond to higher order statistics of the connectivity distribution.

The second order statistics quantify the prevalence of adjacent two edge motifs. Figure 2.1 illustrates the three types of two edge motifs we'll consider.

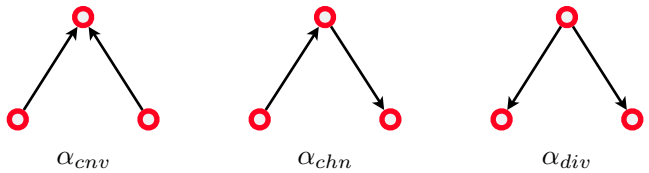


Figure 2.1: **Two edge motifs** The three types of 2-edge motifs we will consider. The α's are quantities associated with each motif. They are defined in (2.2).

The following definitions are essentially the same as in [57]. As in [29], I will absorb the reciprocal motif into the chain motif. Note that in [58], and [57], p and the α's below are marked with hats. Let $\langle d\rangle = pN = |W|/N$ be the mean degree (either in or out). Then we define the second order statistics α by

$$\begin{aligned} p &= \frac{|W|}{N^2} = \frac{1}{N}\left\langle d_{\text{in}}\right\rangle = \frac{1}{N}\left\langle d^{\text{out}}\right\rangle = \frac{\langle d\rangle}{N} \\ \alpha_{\text{cnv}} &= \frac{(\left\langle (d_{\text{in}})^2\right\rangle - \langle d\rangle^2) - \langle d\rangle}{\langle d\rangle^2} \\ \alpha_{\text{div}} &= \frac{(\left\langle (d^{\text{out}})^2\right\rangle - \langle d\rangle^2) - \langle d\rangle}{\langle d\rangle^2} \\ \alpha_{\text{chn}} &= \frac{\left\langle d_{\text{in}} d^{\text{out}}\right\rangle - \langle d\rangle^2}{\langle d\rangle^2} \end{aligned} \tag{2.1}$$

where $\langle\cdot\rangle$ denotes an average over each node of the population. Note that the convergence, and divergence statistics are coefficients of variation for the in and out degree sequences shifted by a factor of $1/\langle d\rangle$. The chain statistic could justifiably be called a coefficient of covariation.

In their original form in [57], the α's are used to quantify deviations of motif statistics from independence. In that sense they are seen to satisfy

$$\begin{aligned} p^2(1+\alpha_{\text{cnv}}) &= \frac{|W^TW| - \text{Tr}(W^TW)}{N(N-1)(N-2)} \approx \frac{|W^TW| - \text{Tr}(W^TW)}{N^3} \\ p^2(1+\alpha_{\text{div}}) &= \frac{|WW^T| - \text{Tr}(WW^T)}{N(N-1)(N-2)} \approx \frac{|WW^T| - \text{Tr}(WW^T)}{N^3} \\ p^2(1+\alpha_{\text{chn}}) &= \frac{|W^2|}{N(N-1)(N-2)} \approx \frac{|W^2|}{N^3}. \end{aligned} \tag{2.2}$$

We can simplify (2.1) by writing them in terms of deviations from the mean degree. Define ϵ_{in} by $d_{\text{in}} = \langle d\rangle + \epsilon_{\text{in}}$, and ϵ^{out} by $d^{\text{out}} = \langle d\rangle + \epsilon^{\text{out}}$. Then we may

express the second order statistics more succinctly as

$$\begin{aligned}\alpha_{\text{cnv}} &= \frac{\langle \epsilon_{\text{in}} \cdot \epsilon_{\text{in}} \rangle}{\langle d \rangle^2} - \frac{1}{\langle d \rangle} \\ \alpha_{\text{div}} &= \frac{\langle \epsilon^{\text{out}} \cdot \epsilon^{\text{out}} \rangle}{\langle d \rangle^2} - \frac{1}{\langle d \rangle} \\ \alpha_{\text{chn}} &= \frac{\langle \epsilon_{\text{in}} \cdot \epsilon^{\text{out}} \rangle}{\langle d \rangle^2}.\end{aligned} \tag{2.3}$$

These expressions emphasize that the second order statistics increase with the variances and covariances between the in and out degree distributions. See figure 3.1 for an illustration of the effects that second order statistics have on the structure of random networks.

2.3 Third order statistics

In the next chapter, I'll construct a random network model for which the third order observed statistics of sample networks are affected by parameters of the model. The definitions of these third order statistics can be found in [20]. These quantities are commonly referred to as *assortativities*. The four types of assortativities we'll consider are illustrated in Figure 2.2. They are defined as the correlation coefficients of the *in* and/or *out* degrees of nodes which are *connected by an edge*. Again, we suppose that we are given a fixed W. In chapter 5, I will show that the prevalence of a certain 3-edge motif ($r(out, in)$) affects the accuracy of a rank one approximation of W, based on the degree sequence.

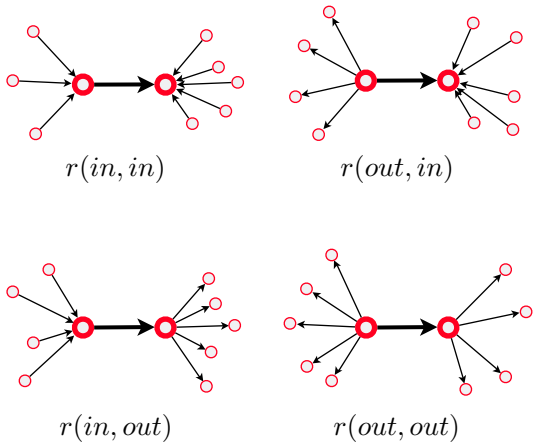


Figure 2.2: **Assortitivities, three edge motifs** The four types of 3-edge motifs we will consider. In each case, these statistics measure the correlations of degrees *across edges*. Hence $r(in, in)$ measures the correlation between the in-degrees of two nodes connected by an edge, $r(out, in)$ measures the correlation between the out-degree of a source node and the in-degree of the target node, $r(in, out)$ measures the correlation between the in-degree of a source node and the out-degree of the target node, and $r(out, out)$ measures the correlation between the out-degrees of two nodes connected by an edge.

Given an edge from node j to node i, I'll refer to node j as a source node, and node i as a target node with respect that edge. Given an edge $W_j^i = 1$, the third order statistics are concerned with four quantities associated with W_j^i: the in and out degrees of the source node j, and the in and out degrees of the target node i. The averages (with respect to the edges) of these four quantities may be written as

$$\begin{aligned}
\bar{\sigma}^{\text{in}} &= \frac{|W^2|}{|W|} \\
\bar{\sigma}^{\text{out}} &= \frac{|WW^T|}{|W|} \\
\bar{\tau}^{\text{in}} &= \frac{|W^TW|}{|W|} = \bar{\sigma}^{\text{out}} \\
\bar{\tau}^{\text{out}} &= \frac{|W^2|}{|W|} = \bar{\sigma}^{\text{in}}.
\end{aligned}$$

Notice that in the above, the letters σ, and τ stand for *source* and *target*.

The four types of third order statistics (assortativities) we'll consider are the following correlation coefficients

$$r(\alpha,\beta) = \frac{\sum_{i,j} W^i_j (d^\alpha_j - \bar{\sigma}^\alpha)(d^\beta_i - \bar{\tau}^\beta)}{\left(\sum_j d^{\text{out}}_j (d^\alpha_j)^2 - |W|(\bar{\sigma}^\alpha)^2\right)^{1/2} \left(\sum_i d^i_{\text{in}} (d^\beta_i)^2 - |W|(\bar{\tau}^\beta)^2\right)^{1/2}}, \tag{2.4}$$

where (α,β) is one of the four pairs (in,in), (in,out), (out,in), or (out,out). Note that the upper index convention for d^i_{in} is ignored in the d^α_i or d^β_i, when α or β is *in*. Equation (2.4) can be simplified by using vector notation. In that case we find

$$r(\alpha,\beta) = \frac{(d^\beta)^T W d^\alpha - |W|\bar{\sigma}^\alpha \bar{\tau}^\beta}{\left(\sum_j d^{\text{out}}_j (d^\alpha_j)^2 - |W|(\bar{\sigma}^\alpha)^2\right)^{1/2} \left(\sum_i d^i_{\text{in}} (d^\beta_i)^2 - |W|(\bar{\tau}^\beta)^2\right)^{1/2}}, \tag{2.5}$$

Finally, we may express the third order statistics in terms of the deviations from the mean degree, $d^\alpha_i = \langle d\rangle + \epsilon^\alpha_i$. With respect to the ϵ's, W, and $\langle d\rangle$ the third order statistics are

$$r(\alpha,\beta) = \frac{(\epsilon^\beta)^T W \epsilon^\alpha - \frac{\left\langle \epsilon^{\text{out}}_i \epsilon^\alpha_i \right\rangle}{\langle d\rangle} \frac{\left\langle \epsilon^i_{\text{in}} \epsilon^\beta_i \right\rangle}{\langle d\rangle}}{\left(\langle d\rangle (\epsilon^\alpha \cdot \epsilon^\alpha) - \frac{\left\langle (\epsilon^{\text{out}}_i \epsilon^\alpha_i)^2 \right\rangle}{\langle d\rangle}\right)^{1/2} \left(\langle d\rangle (\epsilon^\beta \cdot \epsilon^\beta) - \frac{\left\langle (\epsilon^j_{\text{in}} \epsilon^\beta_j)^2 \right\rangle}{\langle d\rangle}\right)^{1/2}}. \tag{2.6}$$

In the next chapter, I will review/present several random network models which will affect the observed statistics presented above.

Chapter 3

Random networks

One of the aims of this thesis is to elucidate the relationship between the structure of a network, and dynamics on the network. Part of the approach I will follow will be to derive expressions which relate the dynamics on a network to statistics of the network. In this endeavor it will be useful to have several random network models which will serve to numerically test the range of applicability of approximations in later chapters.

In this chapter, I will first review the classical random Erdös-Rényi (ER) network model in which edges occur with a set probability p. The rest of the chapter will be devoted to generalizations of the ER model: the SONET model of Nykamp [58], the expected degree model of Chung and Lu [11], and my generalization of the expected degree model. What these models have in common is that they admit parameterizations which affect the expected observed second order statistics. Moreover, we will see that the generalization of the expected degree model is capable of generating networks with a variety of third order statistics. I will also provide heuristics for approximating the spectrum of the expected degree model, and present a representation for the expected degree model and its generalization which is invariant of the number of nodes.

3.1 Review: previous random network models

3.1.1 Erdös-Rényi networks

One of the prototypical random network models is specified by N, the number of nodes, and p the probability of any edge. Random networks are then generated by taking the edges to be independent Bernoulli random variables with probability p. This is widely referred to as the *Erdös-Rényi (ER) model* due to a similar undirected graph model treated in the 1959 paper [17]. However, the model considered in that paper prescribes the number of nodes, and the number of edges (rather then probability of each edge). Apparently, the random graph model in which the probability of each edge is prescribed was first investigated by Gilbert in [23], also in 1959. Nevertheless, I will follow Stigler's law of eponymy and refer to the model where edges occur with some probability as the ER model. This model has been thoroughly studied for the undirected case, e.g. [11]. The random network models we'll consider below are extensions of the ER model in two slightly different directions. All networks referred to in this thesis are directed.

3.1.2 Second order networks (SONETs)

The first generalization of the Erdös-Rényi model that I'll review is the SONET model of Nykamp, and Zhao [57], [58]. In this model, one not only specifies the probability of an edge, but also the joint probabilities of pairs of edges which share a node. In particular, the model is parameterized (at least in the limit of a large number of nodes) by the expected values of the observed second order statistics introduced in section 2.2 (with slight modifications as noted below). Ideally, the SONET model is the maximum entropy distribution such that the marginal distributions on edges and pairs of edges satisfy

$$P(W_j^i = 1) = p$$

$$\begin{aligned} P(W_j^i = 1, W_i^j = 1) &= p^2(1 + \alpha_{\text{rcp}}) \\ P(W_j^i = 1, W_k^i = 1) &= p^2(1 + \alpha_{\text{cnv}}) \\ P(W_j^i = 1, W_j^k = 1) &= p^2(1 + \alpha_{\text{div}}) \end{aligned} \tag{3.1}$$

$$P(W_j^i = 1, W_k^j = 1) = p^2(1 + \alpha_{\text{chn}}). \tag{3.2}$$

Here, α_{rcp} measures the relative frequency of reciprocal edges (edges between two nodes in both directions) compared to the frequency with which reciprocal edges appear in an Erdös-Rényi network. Note, the α_{chn} appearing in (3.2) is different from that defined in section 2.2, in that the definition there includes the count of reciprocal edges. In practice, Nykamp generates SONETs using a dichotomized Gaussian technique [57]. See figure 3.1 for an illustration of the effects of the α parameters on network structure.

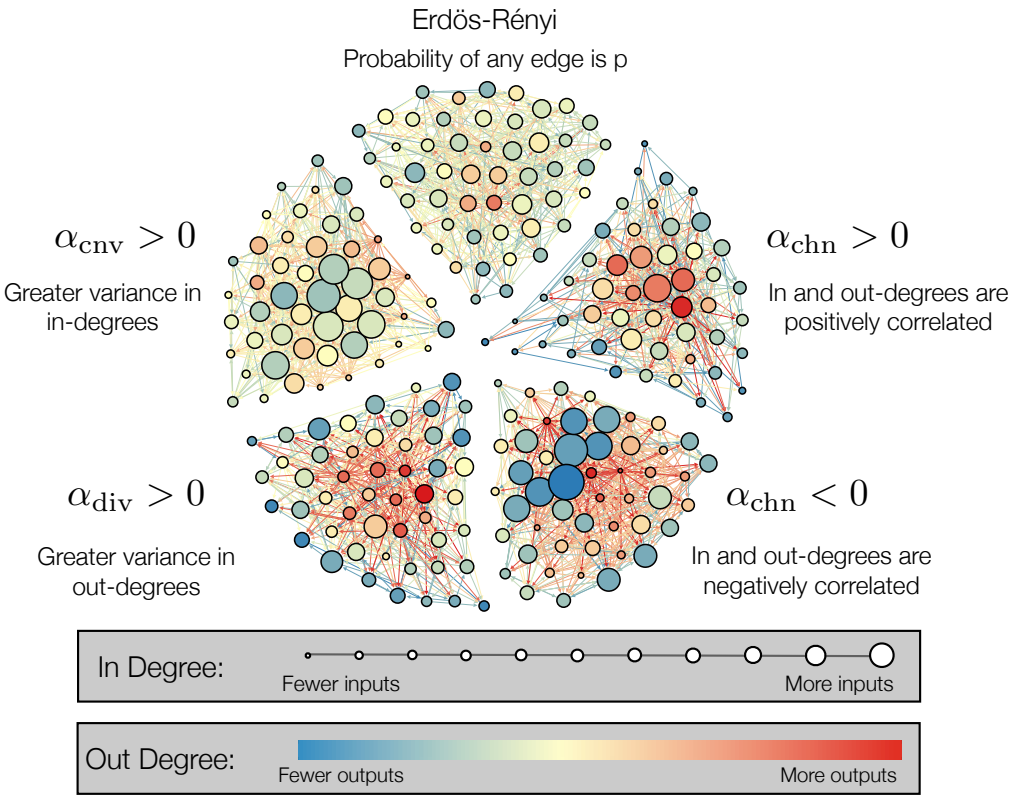


Figure 3.1: **Illustration of SONETS behavior** Sizes of nodes indicate in degrees, while colors indicate out degrees. In the top center we have an example of an ER network: both in and out degrees tend to be somewhat uniform across the network. Increasing α_{chn} results in a greater variance in the in degrees. Similarly, increasing α_{div} results in a greater variance in the out degrees. Changing α_{chn} affects the correlations of in and out degrees.

3.1.3 The expected degree model (EDM)

I first encountered the expected degree model (EDM) in Chung and Lu's book on complex networks [11]. The presentation there is for undirected graphs, but the extension to directed graphs is clear. This is a random network model on N nodes. We start with two nonnegative sequences (which will turn out to be the expected in and out degrees of the nodes) $\boldsymbol{w}_{\text{in}} = \{w^i_{\text{in}}\}$ and $\boldsymbol{w}^{\text{out}} = \{w^{\text{out}}_j\}$, each having values between 0 and $(N-1)$. The $\boldsymbol{w}$ sequences must satisfy

$$\rho := \sum_i w^i_{\text{in}} = \sum_j w^{\text{out}}_j, \tag{3.3}$$

and

$$\max_{i,j} w^i_{\text{in}} w^{\text{out}}_j \leq \rho. \tag{3.4}$$

The probability of an edge from node j to node i is

$$p^i_j = \frac{w^i_{\text{in}} w^{\text{out}}_j}{\rho}. \tag{3.5}$$

Given these probabilities edges are generated independently with their respective probabilities. Following Chung and Lu, let $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$ denote the probability distribution constructed as above. Let W denote a random adjacency matrix distributed as $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$, i.e.

$$P(W^i_j = 1) \;=\; p^i_j. \tag{3.6}$$

The conditions (3.3), and (3.4) are needed to ensure that $p^i_j \in [0,1]$ is a probability. Given $i \in \{1 \ldots N\}$, the meaning of the values w^i_{in}, and w^{out}_i are revealed by considering the expected in and out degrees. Recall, we express the in and out degree of node i as d^i_{in} and d^{out}_i, respectively. For a network W distributed as $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$

we have

$$\begin{aligned}\langle d^i_{\text{in}}\rangle &= \sum_j 1\,P(W^i_j = 1) + 0\,P(W^i_j = 0)\\ &= \sum_j \frac{w^i_{\text{in}} w^{\text{out}}_j}{\rho}\\ &= w^i_{\text{in}} \sum_j \frac{w^{\text{out}}_j}{\rho}\\ &= w^i_{\text{in}}. \end{aligned} \tag{3.7}$$

Similarly, we find $\langle d^{\text{out}}_i\rangle = w^{\text{out}}_i$. Also, observe that ρ is the expected total degree.

We now compute some of the expected observed statistics of a random network $W \sim G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$. The *observed statistics* are just random variables which are functions of W. Within the framework of the expected degree model, we may take expectations of the expressions (2.1) to yield the expected observed statistics

$$\begin{aligned} p &= \frac{\rho}{N^2}\\ \alpha_{\text{cnv}} &= \frac{N}{\rho^2}(\boldsymbol{w}_{\text{in}} \cdot \boldsymbol{w}_{\text{in}} - \rho) - 1\\ \alpha_{\text{div}} &= \frac{N}{\rho^2}(\boldsymbol{w}^{\text{out}} \cdot \boldsymbol{w}^{\text{out}} - \rho) - 1\\ \alpha_{\text{chn}} &= \frac{N}{\rho^2}\left(\boldsymbol{w}_{\text{in}} \cdot \boldsymbol{w}^{\text{out}}\left(1 - \frac{1}{\rho^2}\boldsymbol{w}_{\text{in}} \cdot \boldsymbol{w}^{\text{out}}\right)\right) - 1. \end{aligned}$$

In these expressions "$\cdot$" indicates the dot product.

3.2 An N Invariant Representation for the EDM

By rescaling, and reparameterizing the expected degree sequences we can refer to the network model $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$, reviewed in the previous section, in a way which does not depend directly on N. Specifically, I will show that the expected degree sequences are naturally represented as functions from and to the unit interval. Moreover, this representation yields expressions for the expected observed statistics for a sample network. This may be useful, for example, for considering the behavior of random networks in $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}})$ as $N \to \infty$. Also, this representation yields a

simple way to parameterize families of EDMs, as we will see in the examples below. We relabel the nodes via $i \to i/N$, and define

$$q_{\text{in}}(i/N) := \frac{w^i_{\text{in}}}{N}, \quad \text{and} \quad q^{\text{out}}(j/N) := \frac{w^{\text{out}}_j}{N}.$$

The constraint (3.4) on the $\boldsymbol{w}$'s yields a condition on the q's:

$$\max_{ij} q_{\text{in}}(i/N) q^{\text{out}}(j/N) \leq \frac{1}{N} \sum_k q(k/N), \tag{3.8}$$

where q on the right hand side of (3.8) could be either q_{in} or q^{out}. Notice that the sum on the right hand side of (3.8) is simply a Riemman sum approximation to an integral. With that in mind, we regard q_{in} and q^{out} as functions $q_{\text{in}}, q^{\text{out}} : (0,1] \to [0,1]$, and translate sums like the one above into integrals. To be specific, let's take q_{in} and q^{out} to be continuous functions. From that perspective we can write the condition (3.8) as

$$\max_x q_{\text{in}}(x) \max_y q^{\text{out}}(y) \leq \int q(x) \mathrm{d}x = p. \tag{3.9}$$

(Note: all integrals of q's are over the interval $[0,1]$.) In terms of the q's the probability of an edge from node y to node x is

$$P(W(x,y) = 1) = \frac{q_{\text{in}}(x) q^{\text{out}}(y)}{\int q(x) \mathrm{d}x},$$

As above, we may express the expected values of the observed second order statistics with respect to q_{in} and q^{out}. The resulting expressions are significantly simpler. We have

$$p = \int q_{\text{in}}(x) \mathrm{d}x = \int q^{\text{out}}(x) \mathrm{d}x, \tag{3.10}$$

$$\alpha_{\text{cnv}} = \frac{\int q_{\text{in}}(x)^2 \mathrm{d}x}{p^2} - 1, \tag{3.11}$$

$$\alpha_{\text{cnv}} = \frac{\int q^{\text{out}}(x)^2 \mathrm{d}x}{p^2} - 1, \tag{3.12}$$

$$\alpha_{\text{chn}} = \frac{\int q_{\text{in}}(x) q^{\text{out}}(x) \mathrm{d}x}{p^2} - 1. \tag{3.13}$$

Note that for the EDM, we have that $(\alpha_{\text{chn}}+1)^2 = \alpha_{\text{rcp}}+1$, where α_{rcp} is as in (3.1) . From this we can conclude that the EDM does not encapsulate the SONET model, even though the parameter space for the EDM (the expected degree sequences) is much larger than that for SONET.

By Cauchy-Schwarz we have

$$\left| \int q_{\text{in}}(x) q^{\text{out}}(x) \mathrm{d}x \right| \leq \left(\int q_{\text{in}}(x)^2 \mathrm{d}x \right)^{\frac{1}{2}} \left(\int q^{\text{out}}(x)^2 \mathrm{d}x \right)^{\frac{1}{2}}, \tag{3.14}$$

with equality if, and only if q^{out} and q_{in} are proportional (since we assume they are continuous). Since q_{in} and q^{out} have equal integrals we have that (3.14) is an equality only when $q_{\text{in}} = q^{\text{out}}$. In terms of the expected observed second order statistics (3.14) tells us

$$(\alpha_{\text{chn}} + 1)^2 \leq (\alpha_{\text{cnv}} + 1)(\alpha_{\text{div}} + 1).$$

This is a similar inequality to one previously known [57].

3.2.1 Expected Spectrum

We may write the eigenvalue/eigenvector condition for the adjacency matrix as

$$\lambda f(x) = \sum_x W(x,y) f(y).$$

As an integral we pick up a factor of N

$$\lambda f(x) = N \int W(x,y) f(y) \mathrm{d}y$$

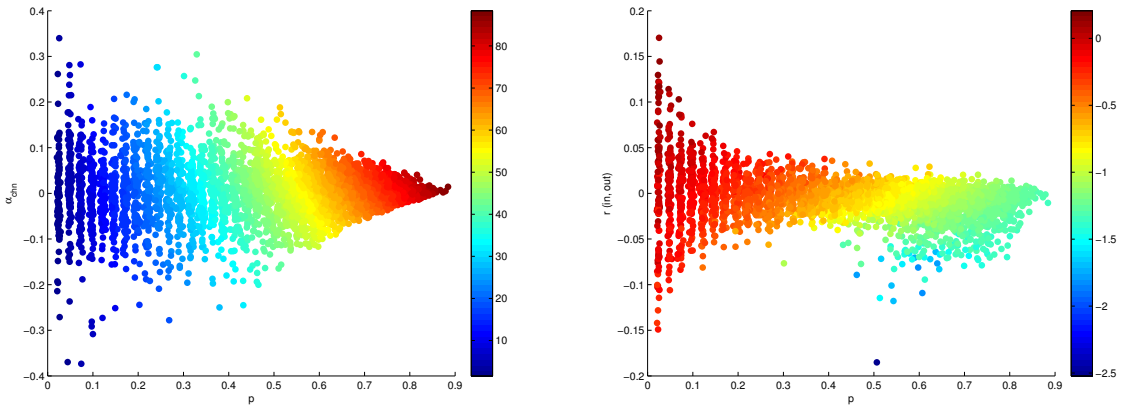
Using the probability of an edge from y to x, we may take expected values to find an equation for an *expected* eigenvalue and eigenvector

$$\begin{aligned} \lambda f(x) &= N \int \frac{q_{\text{in}}(x) q^{\text{out}}(y)}{p} f(y) \mathrm{d}y \\ &= \frac{N}{p} q_{\text{in}}(x) \int q^{\text{out}}(y) f(y) \mathrm{d}y. \end{aligned}$$

Since the only dependence on x on either side is in $f(x)$ on the left, and $q_{\text{in}}(x)$ on the right, we immediately find that we expect $f(x)$ to have values proportional to $q_{\text{in}}(x)$. Taking this for granted, and using the equation for α_{chn} above we may solve for the expected eigenvalue

$$\hat{\lambda}_1 = N(\alpha_{\text{chn}} + 1)p. \tag{3.15}$$

It would seem that this is the only eigenvalue we can predict for this model. Although the argument here isn't rigorous, it appears to agree with the results simulations quite well (see figure 3.2). It is also in agreement with what was found in [42]. Moreover, the figure illustrates that the error is effected by the third order statistic $r(in, out)$, as is also indicated in [42].



(a) (b)

Figure 3.2: **Largest eigenvalue dependency on network statistics.** (a) The largest eigenvalue λ_1 for an ensemble of networks. (b) Relative error between the predicted largest eigenvalue and the actual largest eigenvalue for random networks generated by random EDM models, as in section 3.3.1. The x-axis is the observed statistic p, and the y-axis is the third order observed statistic $r(in, out)$ (see section 2.3). Each dot represents a network, and the color is the difference $\lambda_1 - \hat{\lambda}_1$, where λ_1 is the largest eigenvalue, and $\hat{\lambda}_1$ is the value predicted according to (3.15). These networks had 100 nodes.

I will now present several examples demonstrating the utility of the function representation for the EDM for parameterizing families of EDMs.

Example 3.2.1. Let's take

$$q_{\text{in}}(x) = \beta x^{\gamma}$$

and

$$q^{\text{out}}(x) = \sigma((1-t)x^{\nu} + t(1-x)^{\nu}).$$

The parameter t should be in the interval $[0,1]$. Fixing γ we can find conditions on the other variables to ensure that we get a probability distribution. Thus, we require $\beta \leq \frac{1}{\gamma+1}$. If we wish to have ν as a free parameter, then we must take $\sigma = \beta\frac{\nu+1}{\gamma+1}$ in order to ensure $\int q_{\text{in}} = \int q^{\text{out}}$. With such parameters we have the following expressions for the expected observed second order statistics:

$$p = \frac{\beta}{\gamma+1},$$

$$\alpha_{\text{cnv}} = \frac{\gamma^2}{2\gamma+1},$$

$$\alpha_{\text{div}} = \frac{\nu^2}{2\nu+1} - (1-t)t(1+\nu)^2\left(\frac{2}{2\nu+1} - \frac{\sqrt{\pi}}{4^{\nu}}\frac{\Gamma(\nu+1)}{\Gamma(\nu+\frac{3}{2})}\right),$$

$$\alpha_{\text{chn}} = \frac{\gamma\nu}{\gamma+\nu+1} - t\left(\frac{(\gamma+1)(\nu+1)}{\gamma+\nu+1} - \frac{\Gamma(\gamma+2)\Gamma(\nu+2)}{\Gamma(\gamma+\nu+2)}\right).$$

Here Γ refers to the *gamma* function.

Example 3.2.2. Similarly we could take

$$\begin{aligned} q_{\text{in}}(x) &= \beta x^{\mu} + \delta \\ q^{\text{out}}(x) &= \epsilon x^{\nu} + \zeta \end{aligned}$$

The constraints imposed by (3.8) and (3.9) are now (fixing δ, μ, and ν)

$$\zeta = \delta + \frac{\beta}{1+\mu} - \frac{\epsilon}{1+\nu},$$

$$-\delta \leq \beta \leq 1-\delta,$$

and

$$\max\left(\frac{\nu+1}{\nu}\left(-\delta - \frac{\beta}{1+\mu}\right),\ (\nu+1)\left(\delta + \frac{\alpha}{1+\mu} - 1\right)\right) \leq \epsilon,$$

$$\epsilon \leq \min\left((\nu+1)\left(\delta+\frac{\alpha}{1+\mu}\right)\ ,\ \frac{\nu+1}{\nu}\left(1-\delta-\frac{\beta}{1+\mu}\right)\right).$$

The first and second order statistics can be computed by symbolic software if desired. The resulting expressions are rather lengthy, and do not provide much insight, and so will be omitted.

Example 3.2.3. More generally, suppose that $q_{\text{in}}, q^{\text{out}} : [0,1] \to [0,\infty)$ are two functions with

$$\int q_{\text{in}} = \int q^{\text{out}} = p \in [0,1]$$

Now, given $s,t \in [0,1]$ let

$$q_{\text{in}}^{s} = sq_{\text{in}} + (1-s)p,$$

and

$$q_{t}^{\text{out}} = tq^{\text{out}} + (1-t)p.$$

It is not difficult to show that as long as q_{in}, and q^{out} satisfy the EDM constraint (3.9), q_{in}^{s}, and q_{t}^{out} will also satisfy the EDM constraint. Let α_{cnv}, α_{div}, and α_{chn} be the expected observed second order statistics associated with q_{in}, and q^{out}, and let $\beta_{\text{cnv}}(s,t)$, $\beta_{\text{div}}(s,t)$, and $\beta_{\text{chn}}(s,t)$ be the expected observed second order statistics associated with q_{in}^{s} and q_{t}^{out}. Then we have

$$\beta_{\text{cnv}}(s,t) = s^2\alpha_{\text{cnv}},$$

$$\beta_{\text{div}}(s,t) = t^2\alpha_{\text{div}},$$

and

$$\beta_{\text{chn}}(s,t) = st\alpha_{\text{chn}}.$$

3.3 The generalized expected degree model

Networks generated by the EDM exhibit nearly trivial third order statistics, as figure 3.5 illustrates. Moreover, while the SONET model generates networks which have nontrivial third order statistics, there is no way to control those quantities independently of the second order statistics. In this section, I will introduce a generalization of the expected degree model which is capable of generating network

models which give one control over third order statistics. Having such a model will prove useful in the next chapter where I will investigate the effects of third order statistics on certain low rank approximations to the adjacency matrix.

The inspiration for the generalization of the EDM comes from the singular value decomposition [31], and the fact that the EDM generates edge probabilities via an outer product. The *generalized expected degree model* (GEDM) just adds another outer product (and more if one so desires) to adjust edge probabilities in a way that doesn't change the expected degrees. It should be noted, that in order to leave the expected degrees unchanged, one sacrifices the orthogonality that comes with the singular value decomposition.

As before, let $\boldsymbol{w}_{\text{in}}$, and $\boldsymbol{w}^{\text{out}}$ be the expected degree sequences with

$$\sum_i \boldsymbol{w}^i_{\text{in}} = \sum_j \boldsymbol{w}^{\text{out}}_j = \rho, \tag{3.16}$$

and let $\boldsymbol{u}_{\text{in}}$, and $\boldsymbol{u}^{\text{out}}$ be vectors such that

$$\sum_i \boldsymbol{u}^i_{\text{in}} = \sum_j \boldsymbol{u}^{\text{out}}_j = 0. \tag{3.17}$$

I propose defining the probability of an edge from j to i by

$$P^i_j = \frac{1}{\rho}\boldsymbol{w}^i_{\text{in}}\boldsymbol{w}^{\text{out}}_j + \boldsymbol{u}^i_{\text{in}}\boldsymbol{u}^{\text{out}}_j.$$

The EDM condition (3.4) extends to a condition on the $\boldsymbol{w}$'s, and $\boldsymbol{u}$'s to ensure that P is actually a matrix of probabilities. Namely, $\boldsymbol{w}_{\text{in}}$, $\boldsymbol{w}^{\text{out}}$, $\boldsymbol{u}_{\text{in}}$, and $\boldsymbol{u}^{\text{out}}$ must satisfy the *GEDM condition*

$$0 \leq \min_{i,j}\left(w^i_{\text{in}}w^{\text{out}}_j + u^i_{\text{in}}u^{\text{out}}_j\right) \leq \max_{i,j}\left(w^i_{\text{in}}w^{\text{out}}_j + u^i_{\text{in}}u^{\text{out}}_j\right) \leq 1. \tag{3.18}$$

On the one hand, this generalization of the EDM has room for more general expected degree sequences than the original model. That is, one may now violate the EDM condition (3.4) by having $u^i_{\text{in}}u^{\text{out}}_j < 0$ at i, and j where $w^i_{\text{in}}w^{\text{out}}_j > \rho$. On the other hand, given some analytic expressions for $\boldsymbol{w}_{\text{in}}$, and $\boldsymbol{w}^{\text{out}}$, as in the examples above (formulated there in terms of q's), it may prove difficult to write general expressions

for $\boldsymbol{u}_{\text{in}}$ and $\boldsymbol{u}^{\text{out}}$ which satisfy the conditions (3.16), (3.17), and (3.18).

As mentioned above, using $\boldsymbol{u}$ we can control higher order statistics. To test the effects of third order statistics on some quantity we could choose $\boldsymbol{w}_{\text{in}}$, and $\boldsymbol{w}^{\text{out}}$ such that the EDM condition (3.4) is satisfied, and then vary $\boldsymbol{u}_{\text{in}}$, and $\boldsymbol{u}^{\text{out}}$ in such a way that the GEDM conditions (3.16)-(3.18) are satisfied. Assuming w_{in}, and w^{out} satisfy the EDM condition, the model $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}}, \boldsymbol{u}_{\text{in}}, \boldsymbol{u}^{\text{out}})$ will be guaranteed to satisfy the the GEDM condition provided

$$0 \leq \min_{i,j} w^i_{\text{in}} w^{\text{out}}_j + \min_{i,j} u^i_{\text{in}} u^{\text{out}}_j \leq \max_{i,j} w^i_{\text{in}} w^{\text{out}}_j + \max_{i,j} u^i_{\text{in}} u^j_{\text{in}} \leq 1. \tag{3.19}$$

We can estimate the expected third order statistics (defined in section 2.3) of a network sampled from $G(\boldsymbol{w}_{\text{in}}, \boldsymbol{w}^{\text{out}}, \boldsymbol{u}_{\text{in}}, \boldsymbol{u}^{\text{out}})$. To this end, let us switch notation back to the q functions, as in section 3.2, and let $v_{\text{in}}(x) = u^x_{\text{in}}/N$, and $v^{\text{out}}(y) = u^{\text{out}}_y/N$. The expected observed second order statistics of a matrix in $G(q_{\text{in}}, q^{\text{out}}, v_{\text{in}}, v^{\text{out}})$ are the same as those for a matrix in $G(q_{\text{in}}, q^{\text{out}})$. However, the expected observed third order statistics are nontrivial for $G(q_{\text{in}}, q^{\text{out}}, v_{\text{in}}, v^{\text{out}})$, whereas they are trivial for $G(q_{\text{in}}, q^{\text{out}})$. In particular the expected observed statistic $r(\alpha, \beta)$, where (α, β) is one of the pairs $\{(\text{in}, \text{in}), (\text{in}, \text{out}), (\text{out}, \text{in}), (\text{out}, \text{out})\}$ can be written

$$r(\alpha, \beta) = S^\alpha T^\beta, \tag{3.20}$$

where

$$S^\alpha = \frac{\int v^{\text{out}}(x) q^\alpha(x) \mathrm{d}x}{(p \int q^{\text{out}}(x) q^\alpha(x)^2 \mathrm{d}x - (\int q^{\text{out}}(x) q^\alpha(x))^2 \mathrm{d}x)^2)^{1/2}}, \tag{3.21}$$

and

$$T^\beta = \frac{\int v_{\text{in}}(x) q^\beta(x) \mathrm{d}x}{(p \int q_{\text{in}}(x) q^\beta(x)^2 \mathrm{d}x - (\int q_{\text{in}}(x) q^\beta(x))^2 \mathrm{d}x)^2)^{1/2}}. \tag{3.22}$$

Again, finding analytic expressions for a family of expected in, and out degree sequences (as in examples 3.2.1, 3.2.2, and 3.2.3), *plus* associated adjustments can be prohibitively involved. I have tried, but the GEDM condition (3.18) ends up putting a lot of simultaneous constraints on the parameters of the functions. I will conclude this chapter, however, by detailing a method for generating networks with nontrivial third order statistics using the GEDM.

3.3.1 Generating random, random network models

In this section I will demonstrate a method for constructing random, random network models (i.e. randomly generating random network models). Our aim is to generate a probability matrix P given the number of nodes N, and the average edge probability p. The second and third order statistics of the resulting model will be free to vary. The basic idea of this construction is to simply to generate random expected degree functions, and associated perturbations, and construct P as in the EDM, and the GEDM. The method works like this:

1. Generate two random functions $q_{\text{in}}, q^{\text{out}} : [0,1] \to [0,1]$, scaled so that $\int q_{\text{in}} = \int q^{\text{out}} = p$, if you only want a random EDM, *stop*.

2. Perform a Monte-Carlo optimization to construct mean zero perturbations v_{in} and v^{out} of q_{in} and q^{out} so that for each $x, y \in [0,1]$, $0 \leq q_{\text{in}}(x)q^{\text{out}}(y)/p + v_{\text{in}}(x)v^{\text{out}}(y) \leq 1$.

To generate appropriately scaled random functions I use the following approximation to one dimensional Brownian motion [18],[54]:

$$b(x, \{\omega_i\}) = \frac{\pi}{2\sqrt{2}} \sum_{k=1}^{F} \frac{\omega_k}{k} \sin\left(\frac{\pi}{2} kx\right), \tag{3.23}$$

where for each k, ω_k is distributed as the standard normal distribution, i.e. $\omega_k \sim \mathcal{N}(0,1)$, and F is some predetermined cutoff frequency. Given $c(x) = b(x, \{\omega_i\})$, let $\mu = \min_{x\in[0,1]} c(x)$, and choose $m > 0$. To get a function $q : [0,1] \to [0,1]$, rescale, and translate c via

$$q(x) = \frac{c(x) - \mu + m}{\max_{x\in[0,1]}(c(x) - \mu + m)}. \tag{3.24}$$

The function $q(x)$ is random, and has values between 0 and 1. Examples of such functions, with $m = 0$ are depicted in figure 3.3 for different values of F.

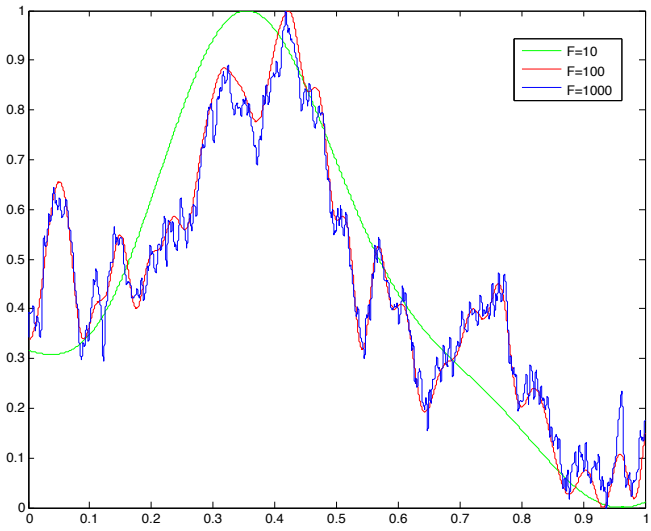


Figure 3.3: **Sample functions** generated as in (3.24), with $m = 0$, and $F = 10$, $F = 100$, and $F = 1000$.

Given $q_0(x)$, and $q_1(x)$ generated according to the above procedure (3.24), and a prescribed p, I set the in and out expected degree densities via

$$q_{\text{in}}(x) = p\frac{q_0(x)}{\int q_0(x)\mathrm{d}t}$$

and

$$q^{\text{out}}(y) = p\frac{c_1(y)}{\int c_1(y)\mathrm{d}t}.$$

Note that as long as p is smaller than the integrals of the q_i's, then the q's as above will take values within $[0, 1]$. To generate the v's, I run a Monte-Carlo optimization for a fixed number of iterations. At each iteration I generate two random functions with mean zero and scaled to have small enough maximum, and minimum values so that the condition $0 \leq q_{\text{in}}(x)q^{\text{out}}(y)/p + v_{\text{in}}(x)v^{\text{out}}(y) \leq 1$ is satisfied. Figures 3.5, and 3.4 depict the statistics of one thousand sample networks generated by this procedure with, $N = 500$, $p = .1$, and $F = 50$.

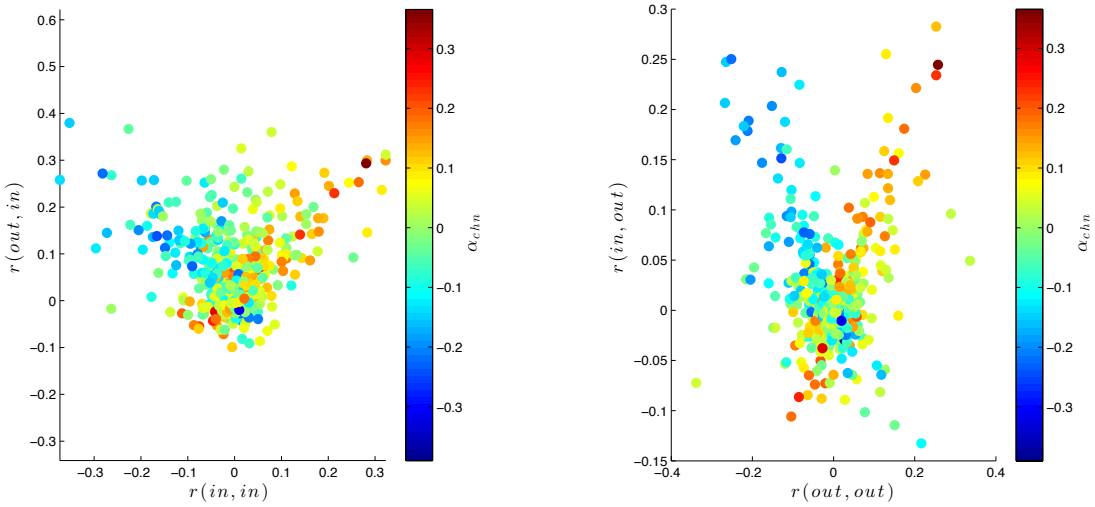


Figure 3.4: **Third order statistics of networks generated by the GEDM.** The color indicates the value of α_{chn}. Each dot corresponds to a network generated according to the procedure in section 3.3.1. This illustrates that third order statistics of the nets generated with the GEDM depend on chains in a nontrivial way.

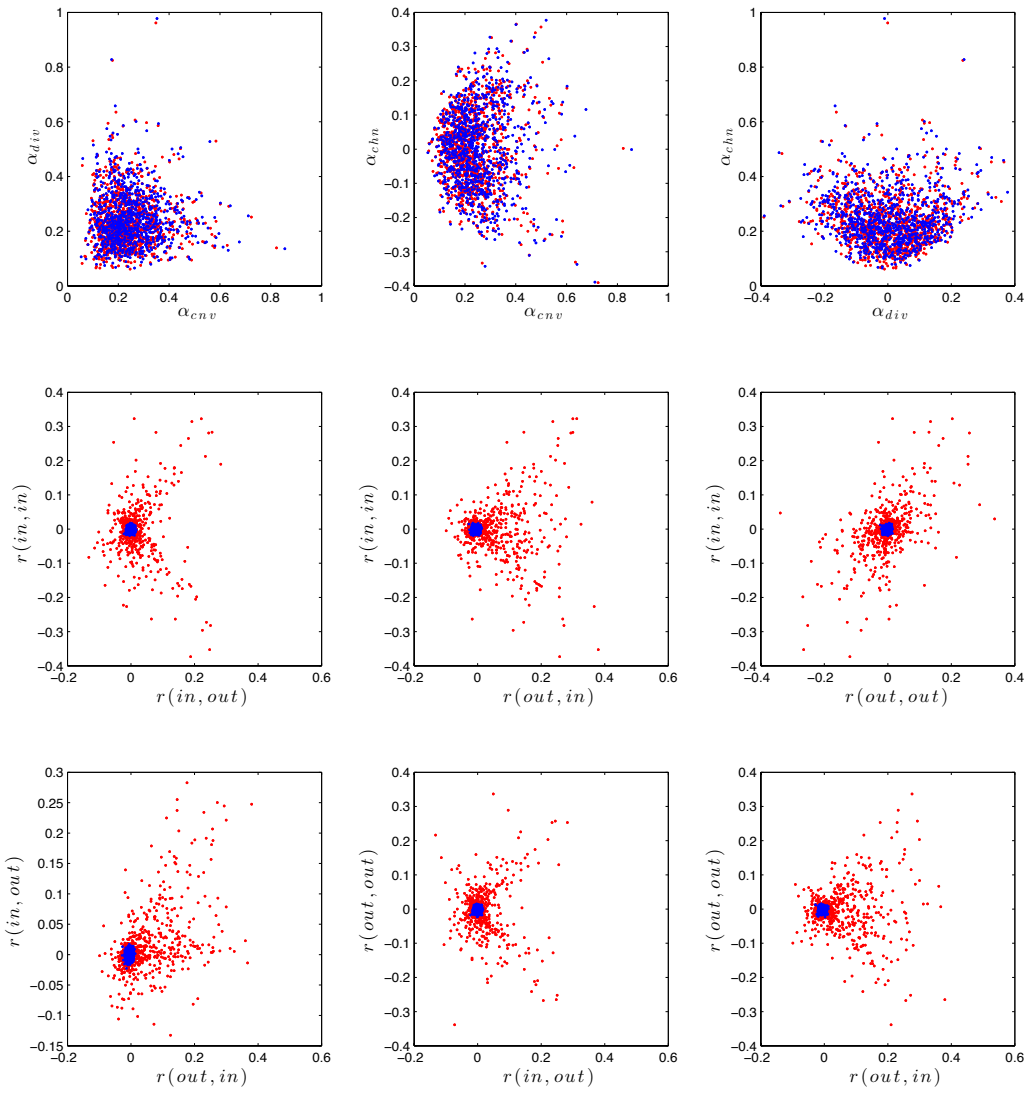


Figure 3.5: **Comparing the statistics for the EDM and GEDM** One thousand networks were generated using the standard EDM (blue), and one thousand networks were generated using the GEDM (red). Each dot corresponds to a network with $N = 500$ nodes, and $p \approx 0.1$. In the first three plots we see that the networks generated with the GEDM, and standard EDM exhibit very similar second order observed statistics. The bottom six plots demonstrate that networks generated with the standard EDM (in blue) consistently have trivial observed third order statistics. On the other hand, the GEDM is capable of generating networks (in red) with a wide variety of third order statistics.

Chapter 4

Patterned networks

Many structures in nature, from trees to mountains to galaxies, seem to be balanced between stochasticity, and symmetry. At a large scale one finds simple geometries, which grow in complexity as one zooms in to smaller scales. At each end of the spectrum one finds a sort of homogeneity. In between, is the domain of complexity and beauty. The brain is no exception to this rule. It is certainly not perfectly symmetric, nor is it random [5]. In the last chapter I presented several random network models. In this chapter I will present a method, which I call *replace and rewire*, for constructing large networks with nontrivial structures. This method is a simple application of the Kronecker product to adjacency matrices. The literature is abound with methods for constructing networks, and use of the Kronecker product for network construction is nothing new. Yet, this approach appears to be novel. It is also both intuitive, and expressive. Applying this method iteratively one can construct a huge variety of large networks, which can range anywhere between total symmetry and complete heterogeneity. The material in this chapter will be used in chapter 8 to illustrate the dynamical reduction method.

After reviewing Kronecker products, and presenting the replace and rewire technique I will provide several examples and discuss their spectral properties. To conclude this chapter I will present a method for constructing networks which have some macro scale structure but which locally look random. This method combines the replace and rewire method with random network models of the previous chapter.

4.1 Background: Kronecker tensor products

Here, I will review the definition of the Kronecker (tensor) product, and Kronecker sum for matrices. The Kronecker product is an essential ingredient for the replace and rewire method I will present below. Along the way, I will also note some of the basic facts concerning the spectra of Kronecker products that I will use in section 4.4 below. There are many references which discuss Kronecker products and their basic properties, for example [2].

Given $A = (a^i_j)$, $m \times n$, and C, $p \times q$ the Kronecker product of A and C is the $mp \times nq$ matrix defined by

$$A \otimes C = \begin{pmatrix} a^1_1 C, & a^1_2 C, & \dots & a^1_n C \\ a^2_1 C, & a^2_2 C, & \dots & a^2_n C \\ \vdots & \vdots & \ddots & \vdots \\ a^m_1 C, & a^m_2 C, & \dots & a^m_n C \end{pmatrix}. \tag{4.1}$$

That is, *$A \otimes C$ consists of $m \times n$ copies of C scaled by the components of A.* A very useful property of Kronecker products is the following. Given A, and C, as above, suppose that B is $n \times r$, and D is $q \times s$, so that the matrix products AB, and CD are well defined. Then we have that

$$(A \otimes C)(B \otimes D) = (AB) \otimes (CD).$$

In particular, if A, and C are square, and we have $Av = \lambda v$, and $Cw = \sigma w$, then

$$(A \otimes C)(v \otimes w) = (Av) \otimes (Cw) = (\lambda v) \otimes (\sigma w) = \lambda\sigma(v \otimes w). \tag{4.2}$$

Hence the eigenvalues of $A \otimes C$ are all of the form $\lambda\sigma$, where λ, and σ are eigenvalues of A, and C, respectively.

One may define the Kronecker sum of two square matrices A, and C by

$$A \oplus C = I_A \otimes C + A \otimes I_C, \tag{4.3}$$

where I_A, and I_C are identity matrices with sizes the same as A, and C. Again, let v, and w be eigenvectors for A, and C, as above. Then

$$(A \oplus C)(v \otimes w) = v \otimes Cw + Av \otimes w = (\lambda + \sigma)v \otimes w. \tag{4.4}$$

Therefore, the eigenvalues of $A \oplus C$ are all of the form $\lambda+\sigma$, with λ and σ, eigenvalues for A, and C.

4.2 Review: Kronecker products for adjacency matrices

Now suppose, that A, and C are adjacency matrices for two networks. What is the graphical interpretation of the network with adjacency matrix $A \otimes C$? Note, we will permit the factor on the *right* of a tensor product, in this case C, to have loops, i.e. edges which start and end at the same node. The factor on the left, A in this case, will be assumed not to have loops, and so the diagonal of A is all zeros. Then from the definition of the Kronecker product we have that $A \otimes C$ is of the form

$$A \otimes C = \begin{pmatrix} \mathbf{0}, & a_2^1 C, & \dots & a_{n-1}^1 C & a_n^1 C \\ a_1^2 C, & \mathbf{0}, & \dots & a_{n-1}^2 C & a_n^2 C \\ \vdots & \vdots & \ddots & \vdots & \\ a_1^{n-1} C, & a_2^{n-1} C, & \dots & \mathbf{0} & a_n^{n-1} C \\ a_1^n C, & a_2^n C, & \dots & a_{n-1}^n C & \mathbf{0} \end{pmatrix}, \tag{4.5}$$

where n is the number of nodes of A, and the $\mathbf{0}$'s along the diagonal each have the same size as C.

Considering (4.5), we can see that

each node of A corresponds to a subpopulation of $A \otimes C$ containing a copy of the nodes in C.

Now suppose, in A, the j^{th} node is connected to the i^{th} node (so $a_j^i = 1$). This means that the j^{th} subpopulation in $A \otimes C$ will be connected to the i^{th} subpopulation. The matrix C then determines how the nodes in the j^{th} population are connected

to the i^{th} subpopulation nodes. An edge $c_l^k = 1$ indicates that the l^{th} node (in subpopulation j) will be connected to the k^{th} node (in subpopulation i). This connectivity corresponds to the $(i,j)^{th}$ block in (4.5). Clearly, if $a_j^i = 0$, then there are no edges from the j^{th} subpopulation to the i^{th} subpopulation and that block is all zeros. Figure 4.1 illustrates the tensor product for two simple networks.

To summarize, in the tensor product $A \otimes C$ of two adjacency matrices, the connectivity of A serves as a macro scale template for the network, while C determines the specific or micro scale inter-connectivity between populations.

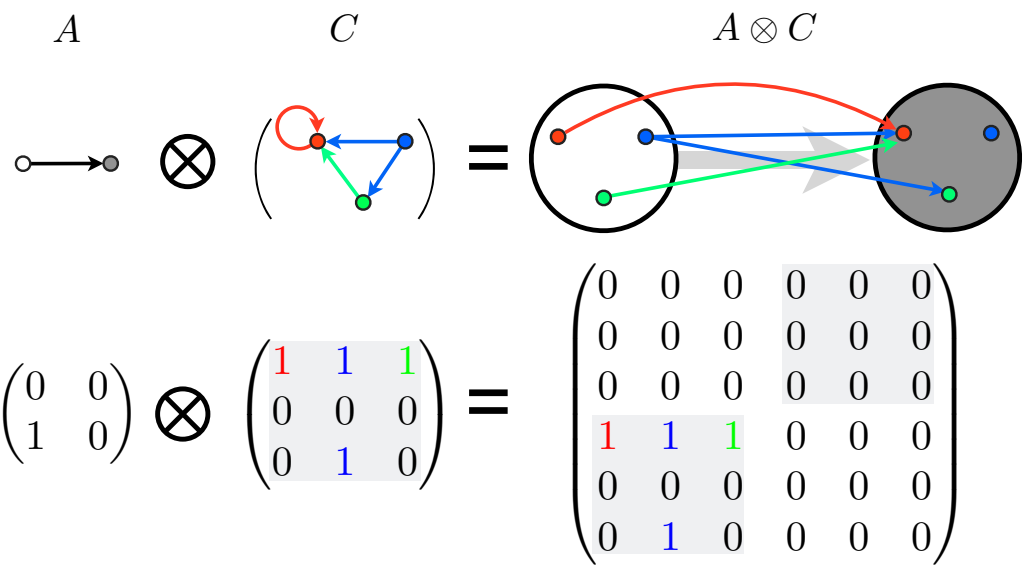


Figure 4.1: **Tensor product of two networks.** Each node of A becomes a subpopulation in the network $A \otimes C$ containing a copy of the nodes of C. The subpopulations are connected along the edges of A. The actual connections from nodes between the subpopulations are determined by the network C. In the this example, all connections are from the first subpopulation onto the second, corresponding to the single edge in A. The edge from the *red* node onto itself in C becomes a connection between the red nodes of the subpopulations. The remaining edges in C are similarly mapped to connections from the first subpopulation onto the second subpopulation, with node indices determined by the edges in C.

4.3 Replace and rewire

I'll now generalize/apply the tensor product just introduced in a manner which is useful for generating large networks with patterned structure. The idea is to combine three networks A, B, and C, where A is $N \times N$, and both B, and C are $K \times K$. As in the product $A \otimes C$, the connectivity of A will determine the *macro* connectivity between subpopulations, and C will determine the specific connectivity *between* subpopulations. The new detail is that each of the subpopulations will have intra-connectivity determined B.

I'll denote this product by $A \otimes_B C$. In terms of the Kronecker product

$$A \otimes_B C = I_A \otimes B + A \otimes C.$$

As a matrix, we have

$$A \otimes_B C = \begin{pmatrix} B, & a_2^1 C, & \dots & a_{n-1}^1 C & a_n^1 C \\ a_1^2 C, & B, & \dots & a_{n-1}^2 C & a_n^2 C \\ \vdots & \vdots & \ddots & \vdots & \\ a_1^{n-1} C, & a_2^{n-1} C, & \dots & B & a_n^{n-1} C \\ a_1^n C, & a_2^n C, & \dots & a_{n-1}^n C & B \end{pmatrix}. \tag{4.6}$$

I call this method *replace and rewire* because we replace the nodes of A with copies of B, and rewire the copies of B along the edges of A according to the connectivity of C. Figure 4.2 illustrates the replace and rewire product.

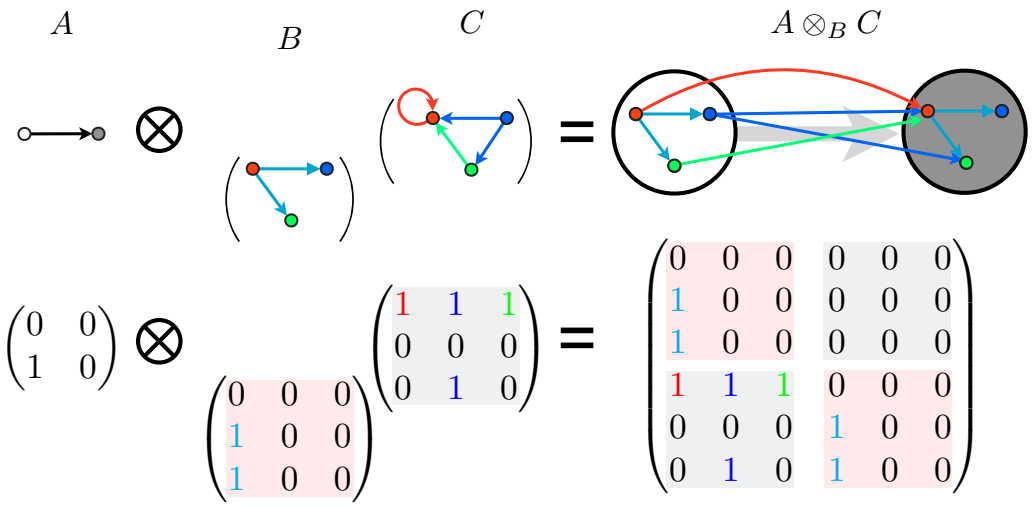


Figure 4.2: **Replace and rewire** This figure is exactly the same as figure 4.1, except each node of A gets a copy of the whole network B in $A \otimes_B C$. Notice that the nodes of B and C are the same (red, green, blue); B gives the connectivity within each copy of the nodes in $A \otimes_B C$, while C gives the connectivity between the copies of the nodes.

Example 4.3.1. I'll now show how to apply the replace and rewire method to construct a lattice, which is embeddable in a 2-torus. Let's call this network/adjacency matrix T. First we must construct an adjacency matrix for a unidirectional ring (see figure 4.3).

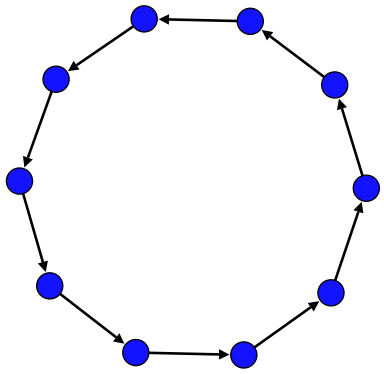


Figure 4.3: **A unidirectional ring.**

The adjacency matrix R of a ring with n nodes can be written as

$$R = \delta^i_{j+1} + \delta^1_n.$$

In words, R is a matrix with ones directly below the diagonal, and a one in the last entry of the first row. To make an undirected ring replace R with $R + R^T$. To construct the adjacency matrix T, we can replace each node of R with a copy of R, and connect neighboring copies so that each node connects with its counterpart in the neighboring population. So we want to replace the nodes of R with copies of R and connect neighbors according to the identity matrix I_R which is the same size as R. As an equation, we have

$$T = R \otimes_R I_R = R \oplus R,$$

where I_R is the identity matrix with the same size as R. Figure 4.4 depicts the network with adjacency matrix T.

In passing, we note that given a unidirectional ring on n nodes, the eigenvalues $\{\omega^i_n\}$ for R are the n^{th} roots of unity. Thus, by (4.4) the eigenvalues for the adjacency

matrix corresponding to the network in figure 4.4 are all of the form

$$\omega_n^i + \omega_n^j.$$

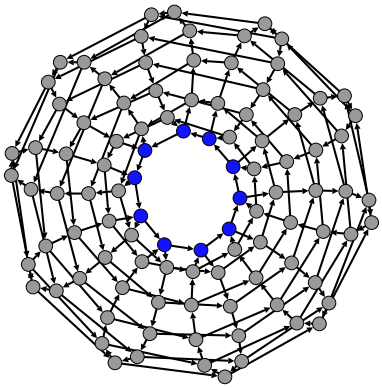


Figure 4.4: **A directed toroidal network** This is the network corresponding to T as constructed above. In this example there are 10^2 nodes.

4.4 Spectra for replace and rewire networks

The eigenvalues/vectors of $A \otimes_B C$ may be subtle, but in some cases we may express them in terms of the eigenvalues/vectors for A, B, and C. In the next several examples, I will demonstrate that for certain forms of A, B, and C, we can completely determine the eigenpairs of $A \otimes_B C$. Each of these examples follow readily from (4.2), and (4.4). Suppose that A has eigenpairs

$$A \quad \longleftrightarrow \quad (\alpha_1, \mathbf{a}_1), \ldots, (\alpha_N, \mathbf{a}_N),$$

B has eigenpairs

$$B \quad \longleftrightarrow \quad (\beta_1, \mathbf{b}_1), \ldots, (\beta_K, \mathbf{b}_K),$$

and C has

$$C \quad \longleftrightarrow \quad (\gamma_1, \mathbf{c}_1), \ldots, (\gamma_K, \mathbf{c}_K).$$

Example 4.4.1. If $B = 0$ then $A \otimes_B C = A \otimes C$, and we can find the eigenpairs as described in (4.2). That is, every eigenpair $(\nu, \mathbf{v})$ of $A \otimes C$ can be expressed as

$$(\nu, \mathbf{v}) = (\alpha_i \gamma_j, \mathbf{a}_i \otimes \mathbf{c}_j),$$

for some i, and j.

Example 4.4.2. The eigenpairs of $A \otimes_B B$ are of the form

$$((\alpha + 1)\beta, \, \mathbf{a} \otimes \mathbf{b}).$$

Example 4.4.3. Suppose that B, and C happen to have a common eigenvector $\mathbf{b}$, with $B\mathbf{b} = \beta\mathbf{b}$, and $C\mathbf{b} = \gamma\mathbf{b}$. Then we're guaranteed that

$$(\alpha\gamma + \beta, \mathbf{a} \otimes \mathbf{b}) \tag{4.7}$$

is an eigenpair for $A \otimes_B C$. I'll refer to this example below when we consider structured random networks.

Example 4.4.4. Note if $C = I_B$ then $A \otimes_B C = A \oplus B$, a case we've already covered. A slightly more complicated case in which we can completely express the eigenpairs of $A \otimes_B C$ in terms of those of A, B, and C is the following. Suppose we have $B = E \otimes F$, and $C = I_E \otimes F$. Then we have

$$\begin{aligned} A \otimes_B C &= A \otimes_{(E \otimes F)} (I_E \otimes F) \\ &= I_A \otimes (E \otimes F) + A \otimes (I_E \otimes F) \\ &= (I_A \otimes E + A \otimes I_E) \otimes F \\ &= (A \oplus E) \otimes F. \end{aligned}$$

If the eigenpairs of E are denoted $(\epsilon, \mathbf{e})$, and those for F by $(\sigma, \mathbf{s})$, then the eigenpairs $(\nu, \mathbf{v})$ of $A \otimes_B C$ each have the form

$$(\nu, \mathbf{v}) = ((\alpha + \epsilon)\sigma, (\mathbf{a} \oplus \mathbf{e}) \otimes \mathbf{s}).$$

In terms of the eigenpairs of A, B, and C we have

$$(\nu, \mathbf{v}) = (\alpha\gamma + \beta, \mathbf{a} \otimes \mathbf{c} + \mathbf{b}). \tag{4.8}$$

One might naively hope that the form of the eigenvalues appearing in (4.8), and (4.7) in terms of those of A, B, and C, might hold in general. The next example shows that such a relationship does not hold in general.

Example 4.4.5. Let

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix},$$

and

$$C = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

The only eigenvalue of B, and C is 0, and the eigenvalues of A are $\{-1, 1\}$. So (4.8) only yields 0 as a guess for the eigenvalues of $A \otimes_B C$. This guess is incorrect, however, since

$$A \otimes_B C = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

which is the adjacency matrix of unidirectional ring with four nodes. Thus the true eigenvalues of $A \otimes_B C$ are the fourth roots of unity.

Example 4.4.6. Lastly, I'd like to mention that applying the replace and rewire method iteratively can result in interesting heterogeneous networks. Figure 4.5 shows a network constructed as follows. Let

$$a = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix},$$

and

$$b = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

For the first step set

$$A_1 = a \otimes_b a.$$

The iteration proceeds

$$A_{n+1} = A_n \otimes_b a.$$

Figure 4.5 shows the network associated with A_8. It is not surprising that this network has fractal like qualities given the iterative procedure which produced it. What may be slightly surprising is the correspondence between the spectrum of A_8 and the filled Julia set associated with the map $f(z) = z^2 - 1$. This Julia set is commonly known as the *basilica set* [1]. Let $f^{(n)}(z)$ be the n^{th} iterate of f, and let $\rho_n(z)$ be the characteristic polynomial of A_n. It can be shown that $\rho_n(z) = f^{(n)}(z)^2$, so this result is not so mysterious after all.

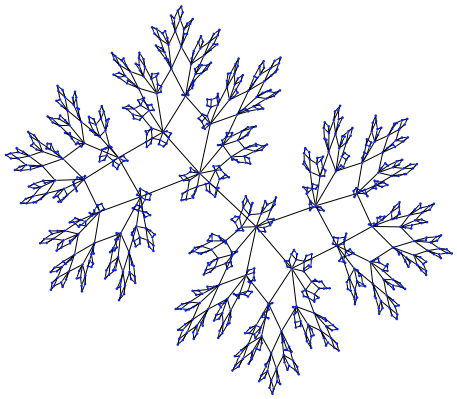


Figure 4.5: **A heterogeneous network constructed using the replace and rewire construction** The network corresponding to the adjacency matrix constructed according to example 4.4.6

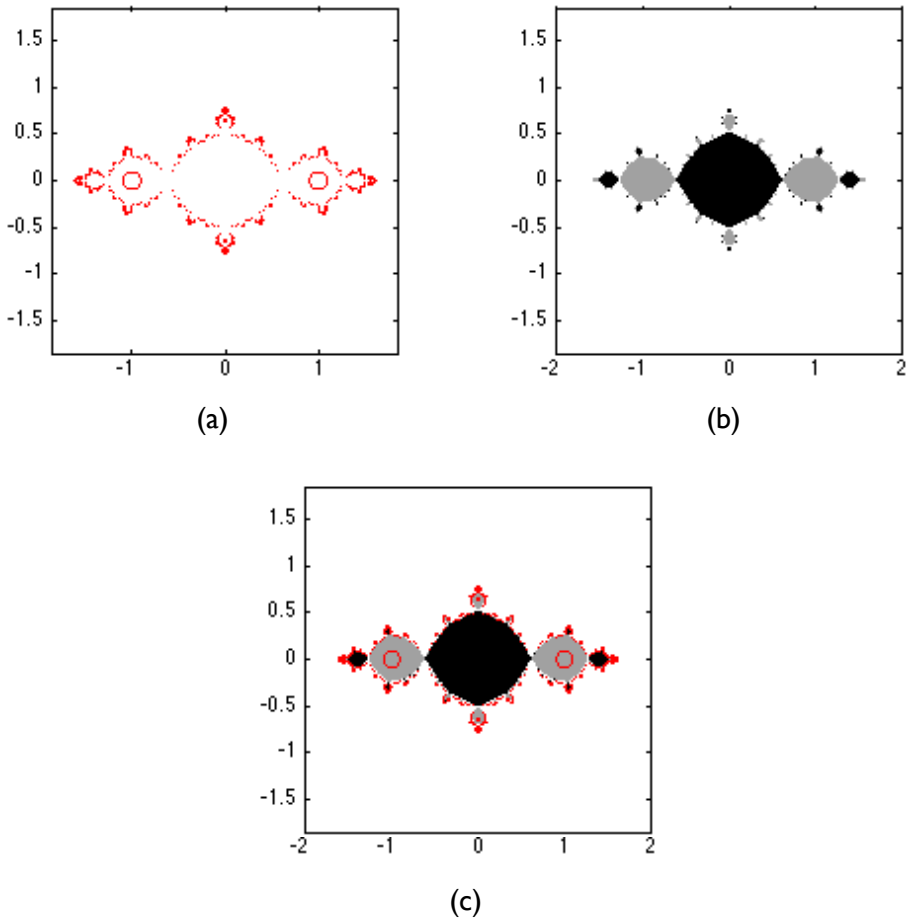


Figure 4.6: (a) The spectrum of the adjacency matrix for the network in figure 4.5. (b) the filled Julia set (the basilica fractal) consisting of points which remain bounded under iteration of the map $f(z) = z^2 - 1$. (c) An interesting correspondence.

4.5 Structured random networks

In this chapter, and chapter 3 we considered two distinct approaches to constructing networks. The networks of this chapters are highly structured, while those of chapter 3 are random. Many real world networks (e.g. brains, social networks, citations, etc.) have structures which are neither perfectly patterned (in the sense of this chapter) nor are they random [38]. In order to investigate the interaction between these regimes, I have combined the random network models of the previous chapter

with the patterned network construction of this chapter. Having a method for constructing networks which are locally random but which have distinctive large scale structure may be useful exploring the role of network structure at different scales.

Previously, there has been extensive research into *stochastic block models* for network generation and model fitting [24]. There has also been work specifically using Kronecker products and probabilistic methods together [34], [33], but the approach there involves taking Kronecker *powers* of a matrix of probabilities. The construction of this section appears to be novel. After detailing the structured/random hybrid model, I will present a conjecture regarding the spectra of networks constructed according to this method. I will then illustrate the conjecture with a numerical example. I conclude the chapter with an example detailing the construction of general networks with multiple types, that will be referred to in section 8.4.

In the replace and rewire technique of the previous section we constructed a network by taking a macro network A, replacing the nodes of A by copies of a network B, and connecting neighboring copies of B according to a network C. The hybrid technique I propose is to take a macro network A, and replace the nodes of A by copies of a $K \times K$ matrix of probabilities P_{rec}, and the edges of A by a $K \times K$ matrix of probabilities P_{ext}. The matrix P_{rec} specifies the probabilities of *recurrent* connections within each subpopulation, while P_{ext} specifies the connections to *external* nodes in neighboring subpopulations. The result is a matrix of probabilities

$$P = A \otimes_{P_{rec}} P_{ext}.$$

A random network W is then generated according to $Prob(W_j^i = 1) = P_j^i$. To indicate W is sampled from the distribution associated with a matrix of probabilities P, I will write $W \sim P$.

A special case of this construction is when the probability matrices P_{rec}, and P_{ext} are constant (except that P_{rec} should be zero along the diagonal to avoid self edges). Let A be an adjacency matrix of a network on N nodes, and let $\mathbb{1}$ be a $K \times K$ matrix of all ones. For probabilities p, and q, define

$$P_A(p, q) = A \otimes_{(p\mathbb{1})} (q\mathbb{1}). \tag{4.9}$$

Even though the specific connections are generated randomly, I put forth the following conjecture. It is inspired by (4.7), and the fact that if the the probability of an edge p for an Erdos-Renyi network is large enough then the principal eigenvector of the adjacency matrix is nearly constant [35].

Conjecture 4.5.1. Let $P_A(p, q)$ be as as above (4.9), and let α be an eigenvalue of A. If p, q, and M are sufficiently large then given $W \sim P_A(p, q)$, the spectrum of W will contain a value (approximately) equal to

$$\alpha qM + pM.$$

If true, the conjecture would tell us that even though the local structure of the network may be random, the macroscopic structure may still be clearly represented in the spectrum of the adjacency matrix.

The proof might be something something like this: with high probability the constant vector is very close to the Perron-Frobenius eigenvector for an Erdös-Réyni network with large enough M, and p [35]. Then by example 4.4.3 we know that since the subpopulations have "the same" eigenvector as the connecting subpopulations, the spectrum will include values of the form (4.7). Since the eigenvalues associated with the constant eigenvectors are very nearly qM, and pM [42] the result follows.

I'll now illustrate the conjecture 4.5.1 with an example.

Example 4.5.2. Let the macroscale structure A be a unidirectional ring with 16 nodes. As noted above, the eigenvalues of A are the 16^{th} roots of unity. Figure 4.7 depicts two random networks constructed according to $P_A(p, q)$. In both cases each subpopulation has 100 nodes. The spectrum of A is clearly represented in both cases. In both instances, we see that the spectrum of $W \sim P_A(p, q)$ includes a ring of values the center of which is shifted relative to the value of p, and scaled relative to the value of q.

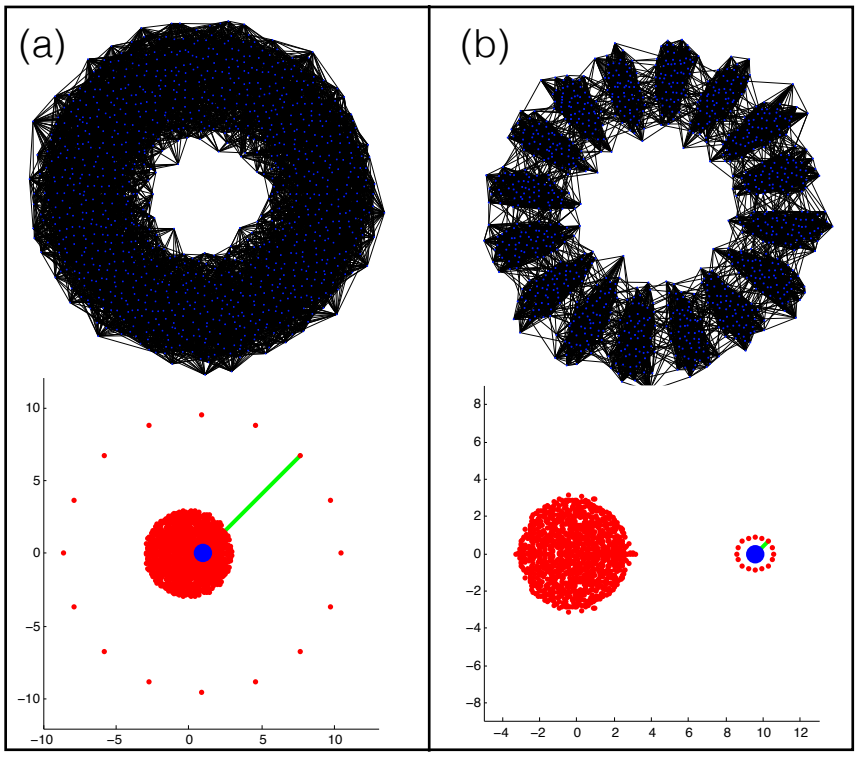


Figure 4.7: **Random rings** Two examples of networks constructed via P in example 4.5.2. (a) $p = .01$, $q = .1$, (b) $p = .1$, $q = .01$. Below each network is its spectrum. The blue dot indicates where the largest eigenvalue of an Erdös-Rényi network would be expected to be where the probability of an edge is p. The length of the green line is equal to the expected value of the largest eigenvalue of an Erdös-Rényi network with edge probability q.

Example 4.5.3. In section 8.4, I will illustrate a dynamical reduction method using a network constructed as in this example. We will have an excitatory population and an inhibitory population. That is, the adjacency matrix will have 1's, 0's, and -1's. A node is excitatory if all of its outgoing edges have positive sign, and inhibitory if all outgoing edges have negative sign. I'll call such a network an *EI network*. In what follows the adjacency matrix A is arbitrary.

The first ingredient for this construction is some macro scale network A. Each node of A will be replaced by *a pair of subpopulations*: an excitatory population of size K_E, and an inhibitory population of size K_I. That is, associated with the i^{th} node of A we will a pair of subpopulations of nodes $(\mathcal{E}_i, \mathcal{I}_i)$. For this model,

we'll specify four probabilities $p_E^E, p_I^E, p_E^I, p_I^I$. The probability of an edge within $\mathcal{E}_i$ will be p_E^E. The probability of an edge from $\mathcal{I}_i$ to $\mathcal{E}_i$ will be p_I^E. The probability of an edge from $\mathcal{E}_i$ to $\mathcal{I}_i$ will be p_E^I. And the probability of an edge within $\mathcal{I}_i$ will be p_I^I. Four more probabilities $q_E^E, q_I^E, q_E^I, q_I^I$ will specify probabilities between neighboring subpopulations. Say there is an edge in A from node j to node i. Then the probability of an edge from a node in $\mathcal{E}_j$ to a node in $\mathcal{E}_i$ will be q_E^E. The other q's follow this pattern. Let $\mathbb{1}_E$, and $\mathbb{1}_I$ be matrices of all ones with sizes $K_E \times K_E$, and $K_I \times K_I$, respectively. Also, let $\mathbf{0}_E$, and $\mathbf{0}_I$ be matrices with all zeros with sizes $K_E \times K_E$, and $K_I \times K_I$, respectively.. I will construct two probability matrices: P_E for the excitatory edges, and P_I for the inhibitory edges. Putting this all together we have

$$P_E = A \otimes \begin{pmatrix} p_e^e \mathbb{1}_E & \mathbf{0}_E \\ p_e^i \mathbb{1}_I & \mathbf{0}_I \end{pmatrix} \begin{pmatrix} q_e^e \mathbb{1}_E & \mathbf{0}_E \\ q_e^i \mathbb{1}_I & \mathbf{0}_I \end{pmatrix},$$

$$P_I = A \otimes \begin{pmatrix} \mathbf{0}_E & p_i^e \mathbb{1}_E \\ \mathbf{0}_I & p_i^i \mathbb{1}_I \end{pmatrix} \begin{pmatrix} \mathbf{0}_E & q_i^e \mathbb{1}_E \\ \mathbf{0}_I & q_i^i \mathbb{1}_I \end{pmatrix}.$$

Let

$$W_E \sim P_E, \qquad \text{and} \qquad W_I \sim P_I.$$

We get an adjacency matrix for an EI network by taking the sum

$$W = W_E - W_I.$$

See example 8.4.1 for a specific instantiation of this model where A is a unidirectional ring with 10 nodes.

Chapter 5

SVD, and low-rank approximations of networks

In the second half of this thesis low rank approximation, and dimension reduction of dynamics will be major themes. The major device we'll employ towards those ends is the *singular value decomposition* (SVD) of the adjacency matrix of a network. In some sense, the SVD gives the best decomposition of a matrix. If we think of the matrix as a linear transformation, the SVD tells us which vectors get changed the most. If the matrix is viewed as an array of data, the SVD picks out the most dominant features. After reviewing some facts about SVD, most of which can be found in [25], and some of the previous work regarding applications of SVD to networks, I will supply numerics and heuristics to support the claims that for some random network adjacency matrices the normalized degree sequences are approximately equal to the principle singular vectors, and yield an accurate prediction of the largest singular value. I will show that the accuracy of these predictions is correlated with one of the third order observed statistics reviewed in section 2.3. The material in this chapter will be used as part of the justification for a dimension reduction method in chapter 8.

For an excellent primer on the SVD, see [31]. Many theorems regarding SVD in general including perturbation theory, can be found in the classic text by Golub and Van Loan, [25].

5.1 Review of basic facts regarding SVD & networks

In this section, I'll briefly review the SVD, and some of its basic properties in the context of adjacency matrices. The SVD of an arbitrary real matrix A is typically expressed as

$$A = U\Sigma V^T,$$

where the matrices U, and V are orthogonal matrices, and Σ is diagonal with entries $\Sigma^i_j = \delta^i_j \sigma_i$. The *singular values* σ_i are ordered so that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N \geq 0$. The columns of U, and V are referred to as the *left*, and *right singular vectors*, respectively. If needed, I will write the k^{th} singular value of a matrix A as $\sigma_k(A)$ to avoid ambiguity. It may be of interest that the singular values of A are the square roots of the eigenvalues of A^TA (or, equivalently AA^T), and that the left and right singular vectors are the eigenvectors of AA^T and A^TA, respectively. There is also an elegant geometric characterization of the singular values. Namely, if we think of A as linear transformation than the singular values are exactly the lengths of the semi-axes of the ellipsoid resulting from applying A to the unit sphere.

To list further properties of the SVD, we will need to define several matrix norms. For a matrix A the *Frobenius norm* of A is

$$\|A\|_F = \sqrt{\sum_{i,j} (A^i_j)^2}.$$

The *p-norm* of A is given as the operator norm

$$\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p},$$

where for a vector v, $\|v\|_p$ is the standard vector p-norm.

As before, given an adjacency matrix of a network (digraph) W, let d_{in}, and d^{out} be the vectors of in and out degrees, and let $|W|$ denote the total number of edges in the network. Then we have

$$\|W\|_F = \sqrt{\sum_{i,j} |W^i_j|} = \sqrt{|W|},$$

$$\|W\|_1 = \max_j \sum_i W_j^i = \max d^{\text{out}},$$

and

$$\|W\|_\infty = \max_i \sum_j W_j^i = \max d_{\text{in}}.$$

Let $W = U\Sigma V^T$ be the SVD of W. The singular values, and matrix norms of W are related in some nice ways. For instance,

$$\sigma_1(W) = \|W\|_2, \tag{5.1}$$

and

$$|W| = \|W\|_F^2 = \sigma_1^2 + \cdots + \sigma_N^2. \tag{5.2}$$

Define the *rank-k approximation* of W by

$$W_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T, \tag{5.3}$$

where u_i and v_i are the i^{th} column vectors of U, and V. Then for $k = 1, \ldots, N-1$

$$\min_{\text{rank}(V)=k} \|W - V\|_2^2 = \|W - W_k\|_2^2 = \sigma_{k+1}^2. \tag{5.4}$$

Equation (5.4) tells us that W_k is the best *rank-k* approximation to W.

5.2 Prior work - SVD & Discrepancy

A number of applications of SVD to networks are concerned with automatically detecting community structures. Of more interest to us, however, are the results concerning discrepancy. In his thesis [8], and two papers [6],[7] Butler shows that the *discrepancy* (according to him "a measurement of how randomly edges have been placed," [7]) of a given network is closely related to the second singular value of its normalized adjacency matrix. Even though this result isn't exactly what I need to justify later approximations, it is closely enough related to warrant a quick review.

Given an adjacency matrix W, define

$$P = \frac{d_{\text{in}} d^{\text{out}}}{|W|}. \tag{5.5}$$

Recall that I am following the convention (see section 2.1) that $d_{\text{in}} d^{\text{out}}$ is an outer product, and thus P is a matrix. If one views W as a random matrix conditioned on its degree sequence, then one could think of P_j^i as an approximation to the probability of an edge from node j to i. (This interpretation breaks down in some cases, as P_j^i can be larger than 1.) The discrepancy is a measure of how much W differs from P.

Suppose the network W has no sinks or sources. That is, say $d_{\text{in}}^i, d_i^{\text{out}} > 0$, for every i. Also, say $X \subset \{1, \ldots, N\}$ is a subset of nodes. Let ψ_X be the vector which is one at indices in X, so ψ_X is the *indicator* vector of X, and define the degree functions

$$d_{\text{in}}(X) = \sum_{i \in X} d_{\text{in}}^i,$$

and

$$d^{\text{out}}(X) = \sum_{i \in X} d_i^{\text{out}}.$$

Notice that given $X, Y \subset \{1, \ldots, N\}$ we have that the number of edges from nodes in X to nodes in Y is

$$E(X \to Y) = \psi_Y^T W \psi_X.$$

Let

$$\alpha(X, Y) = \frac{1}{\sqrt{d^{\text{out}}(X) d_{\text{in}}(Y)}} |\psi_Y^T (W - P) \psi_X|.$$

The function α measures the difference in the number of edges from X to Y versus the expected number of edges from X to Y if all that is known is the degree of node. The *discrepancy* of W is

$$\text{disc}(W) = \max_{X, Y \subset \{1, \ldots, N\}} \alpha(X, Y).$$

The discrepancy is always between 0 and 1, due to the factor of $\frac{1}{\sqrt{d_{\text{in}}(Y) d^{\text{out}}(X)}}$ in the definition of α. The discrepancy of an Erdös-Réyni network is small. On the

other hand, the discrepancy of a unidirectional ring with N nodes, (as in example 4.3.1) is $1 - \frac{1}{N}$. Calculating the discrepancy for an arbitrary network appears to be combinatorially prohibitive, making it a somewhat impractical measure for large networks. On the other hand, the discrepancy is bounded by a singular value which is simple to compute.

Butler shows ([8],[6]) that

$$\text{disc}(W) \leq \sigma_2(\hat{W}) \leq 150\text{disc}(W)(1 - 8\log(\text{disc}(W))), \tag{5.6}$$

where

$$\hat{W}^i_j = \frac{W^i_j}{\sqrt{d^i_{\text{in}} d^{\text{out}}_j}} \tag{5.7}$$

is the normalized adjacency matrix. The expression (5.6) is interesting as it relates the second singular value of $\hat{W}$ with how much W differs from the rank one approximation given by P. Moreover, owing to the form of $\hat{W}$, one can show that its best rank one approximation is given by $\hat{P}$ defined by

$$\hat{P}^i_j = \frac{\sqrt{d^i_{\text{in}} d^{\text{out}}_j}}{|W|}.$$

Therefore $\sigma_2(\hat{W}) = \sigma_1(\hat{W} - \hat{P}) = \|\hat{W} - \hat{P}\|_2$ (as in (5.4)).

Butler's result is close to my claim that for a network which has small higher order observed statistics the principle singular vectors are close to the normalized in and out degree vectors. However, it is not obvious how the singular vectors of W relate to those of $\hat{W}$, nor how the statistics of the network would impact $\sigma_2(\hat{W})$. I will provide further (numerical) evidence that P is a good rank-1 approximation to W when the degree sequences are the most prominent structural feature of a network in the next section.

5.3 Observations regarding the SVD and degree sequences for random matrices

In chapter 8, I will introduce a dimension reduction method for dynamics on a network which relies on the substitution $W \to P$. Below, I will provide support to the claim that for networks which do not have significant correlations in the structure beyond the degree sequences, the matrix P defined by (5.5) is a good rank-1 approximation to W, and that $\sigma_1(P) \approx \sigma_1(W)$. This support will mainly be in the form of numerical results, and a heuristic argument for $\sigma_1(P) \approx \sigma_1(W)$.

Note: it is clear that P should be an excellent rank-1 approximation to W if W is essentially random except for the structure contained in P (the degree sequences). On the other hand, the precise relationship between the statistical structure of a network, and its SVD is not obvious, and will require further research to clarify. The evidence presented here is meant as a first step in that direction. Throughout this section I assume that *all higher order correlations (beyond the third order statistics) are negligible.*

Figure 5.1 illustrates the type of agreement that one finds between the principle singular vectors and the in and out degrees.

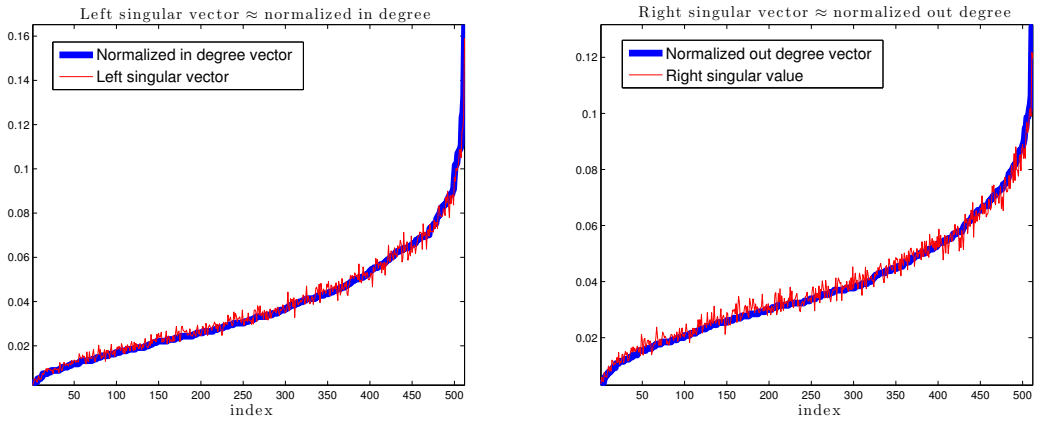


Figure 5.1: **Typical agreement between singular vectors and degree sequences** This is a generic example showing the strong agreement one finds between the singular vectors (red) and degree sequences (blue). The indices in each plot have been arranged so the respective degree vectors are strictly increasing. While the fit is not perfect, it is qualitatively quite close. In this instance the adjacency matrix was generated using the SONET model with parameters $N = 512$, $p = 0.1$, $\alpha_{\text{rcp}} = 0$, $\alpha_{\text{cnv}} = 0.4$, $\alpha_{\text{div}} = 0.3$, and $\alpha_{\text{chn}} = .2$.

First I will consider P as a rank-1 approximation to W. Then I will show that the largest singular value of W is well approximated by the largest singular value of P. In both instances, we will see that the extents to which these approximations hold are correlated with the third order observed statistic $r(out, in)$ introduced in section 2.3. Recall, given a network, $r(out, in)$ is the correlation across all edges between the out degrees of the source nodes, and the in degrees of the target nodes. The three models I will consider here will be those introduced in chapter 3: the SONET model, the expected degree model (EDM), and the generalized expected degree model (GEDM).

5.3.1 P is a good rank-1 approximation of W

As above, P is defined by

$$P = \frac{d_{\text{in}} d^{\text{out}}}{|W|}. \tag{5.8}$$

The claim that P *is a good rank-*1 *approximation of* W requires qualification. Let $W_1 = \sigma_1 u_1 v_1^T$ be the best rank-1 approximation to W associated with the SVD.

For the EDM, and the SONET model, one finds consistently that $P \approx W_1$, but the approximation for the GEDM is more subtle. What I have found is that generally, as long as $r(out, in)$ is small we have $||P - W||_2 \approx \|W_1 - W\|_2$, and that approximation gets worse as $r(out, in)$ deviates from 0. To support this claim, we need an appropriate metric to measure the relative quality of the approximation $\|W - P\|_2$ compared with $\|W - W_1\|_2$.

Let $J = \{i_1, \dots i_k\}$ be any set of indices between 2 and N (where W is $N \times N$) let

$$W_J = \sum_{i \in J} \sigma_i u_i v_i^T.$$

Then, since $1 \notin J$, and σ_1 is the largest singular value we find (as in (5.1))

$$\|W - W_J\|_2 = \left\| \sum_{k \notin J} \sigma_k u_k v_k^T \right\|_2 = \sigma_1.$$

Also, equation (5.4) tells us that

$$\|W - P\|_2 \geq \|W - W_1\|_2 = \sigma_2.$$

Assume that $\sigma_2 < \sigma_1$. Then putting the above facts together, the measure that I propose is

$$\rho_1(P) := \frac{\|W - P\|_2 - \sigma_2}{\sigma_{gap}}, \tag{5.9}$$

where

$$\sigma_{gap} = \sigma_1 - \sigma_2 \tag{5.10}$$

is the *singular gap*. I will call $\rho_1(P)$ the *relative rank-1 error* of P to W . The relative rank-1 error $\rho_1(P)$ measures how much greater $\|W - P\|_2$ is than $\|W - W_1\|_2$, relative to the best a rank-1 approximation to W could be if it were restricted to have no component in the W_1 direction. Since $\sigma_2 > \sigma_1$, we have $\rho(A) = 0$, if, and only if $A = W_1$, and $\rho(W_J) = 1$ if $1 \notin J$.

In figure 5.2, we see $\rho_1(P)$ for networks sampled from EDMs. The purpose of including this plot is simply to show that $\rho_1(P)$ is very small for networks sampled from EDMs. I didn't include any plots comparing $\rho_1(P)$ with network statistics in this case because none of the network statistics were significantly correlated with

$\rho_1(P)$.

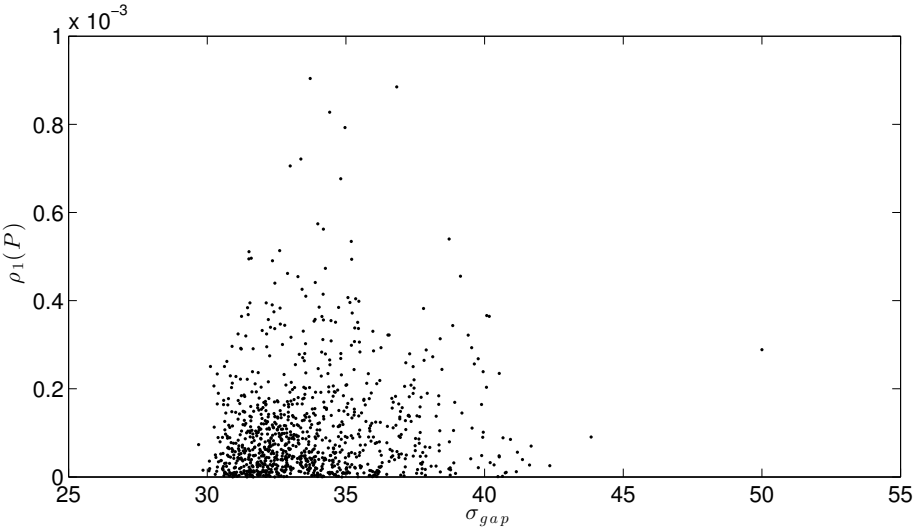


Figure 5.2: **Relative rank-1 error for the EDM** For this plot, 1000 networks were sampled from random EDMs, as in section 3.3.1, with $N = 400$, and $p = 0.1$. Notice that the y-axis is scaled by 10^{-3}. For networks sampled from the expected degree model one generally finds very good agreement between P and W_1. The second order statistics of the sampled networks varied as follows: $-0.338 \leq \alpha_{\text{chn}} \leq 0.312$, $0.02 \leq \alpha_{\text{cnv}} \leq 0.543$, and $0.02 \leq \alpha_{\text{div}} \leq 0.683$. None of the statistics showed significant correlations with the relative rank-1 error.

In the following figures 5.3, and 5.4 we see various plots depicting the value of $\rho_1(P)$ as a function of the observed statistics reviewed in chapter 2. In each one, the statistic which shows strongest correlation with $\rho_1(P)$ is $r(out, in)$, and we see that the relative rank-1 error increases with $|r(out, in)|$.

The numerics of this section helped inform our assumptions for the application of the drive reduction introduced in chapter 8. They also indicate an interesting relationship between the SVD and network statistics, that may warrant further research.

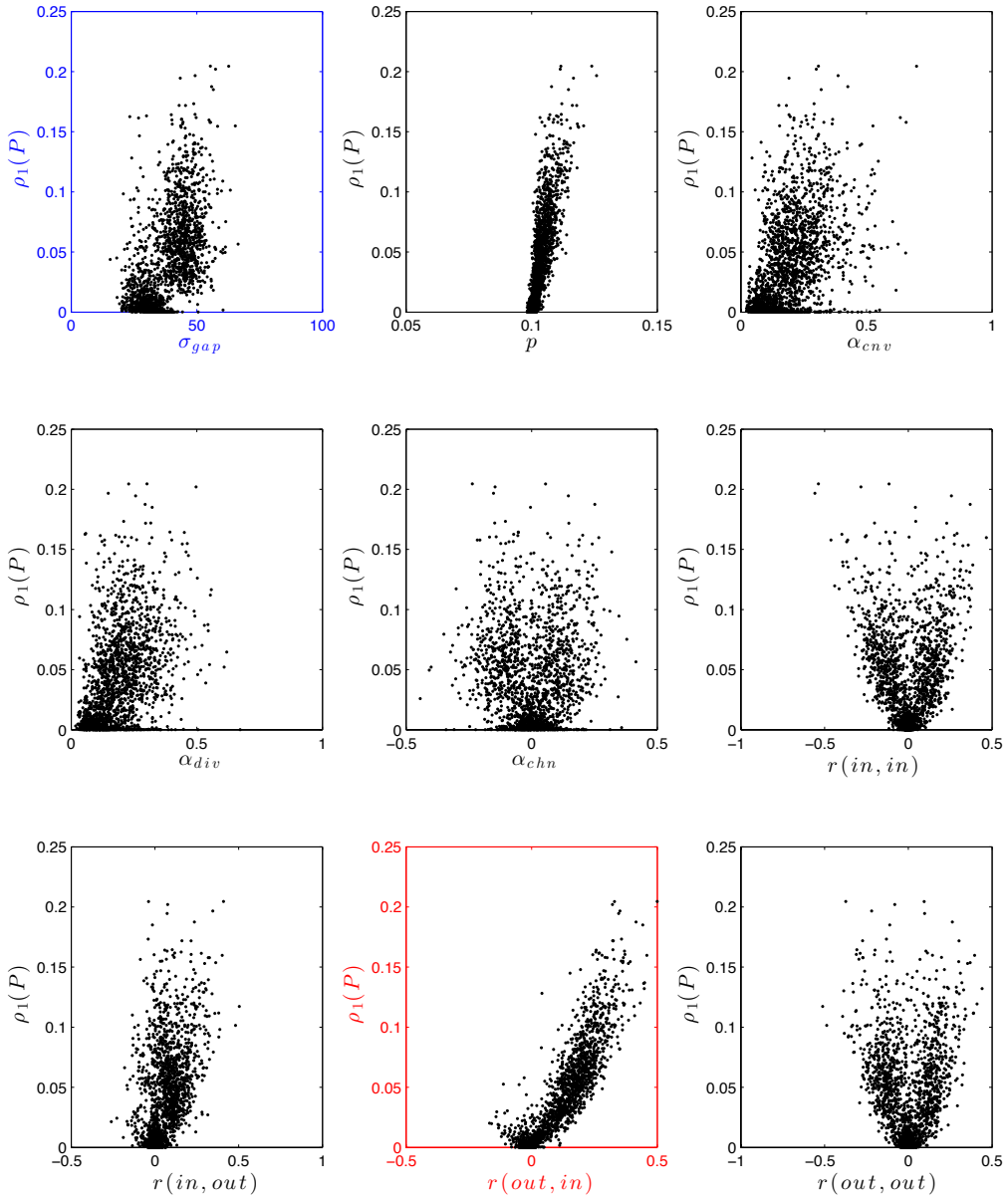


Figure 5.3: **Comparing the relative rank-1 error and network statistics for the GEDM** One thousand networks were generated with $N = 400$, and $p \approx 0.1$, using the GEDM as in example 3.3.1. In every plot, each data point represents one network. The top left (blue frame) plot shows the relative rank-1 error $\rho_1(P)$ along the y-axis, and the singular gap along the x-axis, for reference. The remaining plots have a network statistic along the x-axis and $\rho_1(P)$ along the y-axis. Of particular interest is the bottom middle plot (red frame), which shows that $\rho_1(P)$ tends away from 0 with $r(out, in)$. The other statistics are included for reference to show that they do not correlate as strongly with $\rho_1(P)$ as $r(out, in)$.

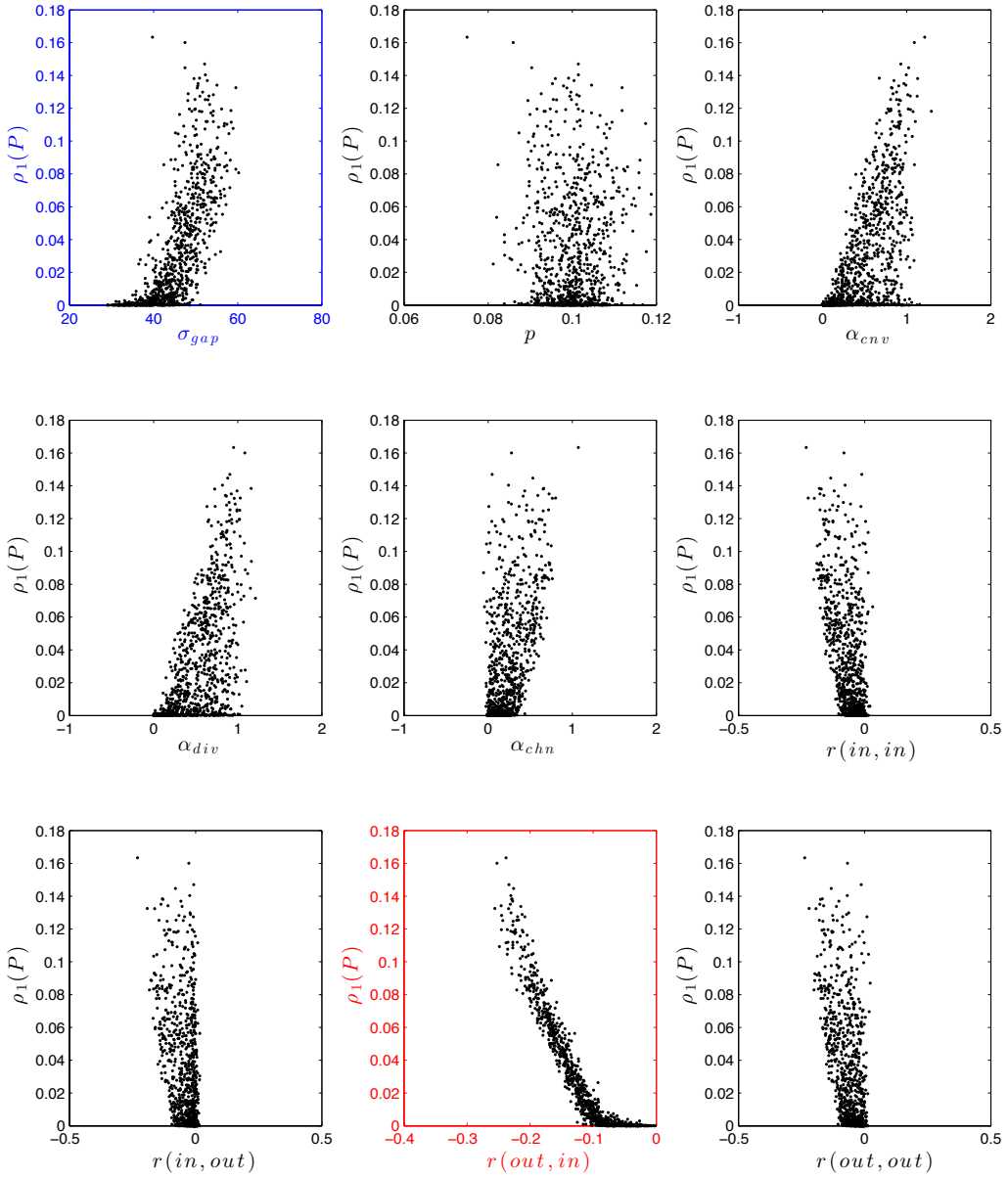


Figure 5.4: **Comparing the relative rank-1 error and network statistics for SONETs** One thousand networks were generated with $N = 400$, and $p \approx 0.1$, and various second order parameters using the SONET model reviewed in section 3.1.2. In every plot, each data point represents one network. The top left (blue frame) plot shows the relative rank-1 error $\rho_1(P)$ along the y-axis, and the singular gap along the x-axis for reference. The remaining plots have a network statistic along the x-axis and $\rho_1(P)$ along the y-axis. As with figure 5.3 above, the bottom middle plot (red frame) shows that $\rho_1(P)$ tends away from 0 with $r(out, in)$.The other statistics are included for reference to show that they do not correlate as strongly with $\rho_1(P)$ as $r(out, in)$.

5.3.2 Approximating $\sigma_1(W)$ with $\sigma_1(P)$

In this section, I will show through numerics, and heuristics that the principal singular value, $\sigma_1(W)$ of W is well approximated by the 2-norm of P, i.e. $\sigma_1(P)$. We have

$$\sigma_1(P) = \|P\|_2 = \frac{\|d_{\text{in}}\|_2 \|d^{\text{out}}\|_2}{|W|},$$

where the norms of the in and out degree vectors are standard Euclidean norms. Figures 5.6, 5.7, and 5.8 show the differences between the largest singular value σ_1 of W, and $\sigma_1(P)$ for the three random network models we have been considering. In all three figures we see $\sigma_1(W) - \sigma_1(P)$ tends away from 0 with $r(out, in)$.

5.3.3 Supporting heuristics

I can provide some intuition for why the accuracy of the approximation $\sigma_1(W) \approx \sigma_1(P)$ is affected by the third order statistic $r(out, in)$. As mentioned above $\sigma_1 = \sqrt{\lambda_1}$, where λ_1 is the largest eigenvalue of both $W^T W$ and $W W^T$. Let's assume that when $r(out, in) \approx 0$, the principal singular vectors are well approximated by the normalized degree sequences. It is well known that for a symmetric real matrix A, the largest eigenvalue is the maximum of the Rayleigh quotient

$$\max_{u \neq 0} \frac{u^T A u}{u^T u}. \tag{5.11}$$

With the degree sequence approximation we have

$$\sigma_1^2 \approx \frac{d^{\text{out}} W^T W \cdot d^{\text{out}}}{\|d^{\text{out}}\|_2^2} = \frac{1}{\|d^{\text{out}}\|_2^2} \sum_{i,j,k} d_i^{\text{out}} (W_i^k W_j^k) d_j^{\text{out}}. \tag{5.12}$$

The sum on the right hand side of (5.12) is pictured in figure 5.5.

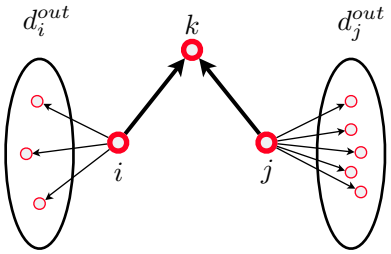


Figure 5.5: **The sum in the Rayleigh quotient** (5.12) is over all i, j, k of this form.

Say $r(out, in) > 0$, and say $d_i^{\text{out}} > \langle d \rangle$. If $W_i^k = 1$ (i.e. there is an edge from i to k), then $r(out, in) > 0$ increases the likelihood that $d_{\text{in}}^k > \langle d \rangle$. Going the other way, if $d_{\text{in}}^k > \langle d \rangle$, and there is an edge from j to k, then $r(out, in) > 0$ increases the chances that $d_j^{\text{out}} > \langle d \rangle$. Therefore $r(out, in) > 0$ the products of the out degrees in (5.12) , and thus we would expect λ_1 to be larger when $r(out, in) > 0$, then when $r(out, in) = 0$. A similar analysis of the case $r(out, in) < 0$ indicates that $r(out, in) < 0$ should lead to a lower value of λ_1 then for when $r(out, in) = 0$.

In [42] they make a bit more rigorous of an argument to show that for *Markovian* networks, the third order statistic $r(in, out)$ affects the largest eigenvalue of W. It may be possible to make these arguments somewhat more rigorous by following the argument they make therein.

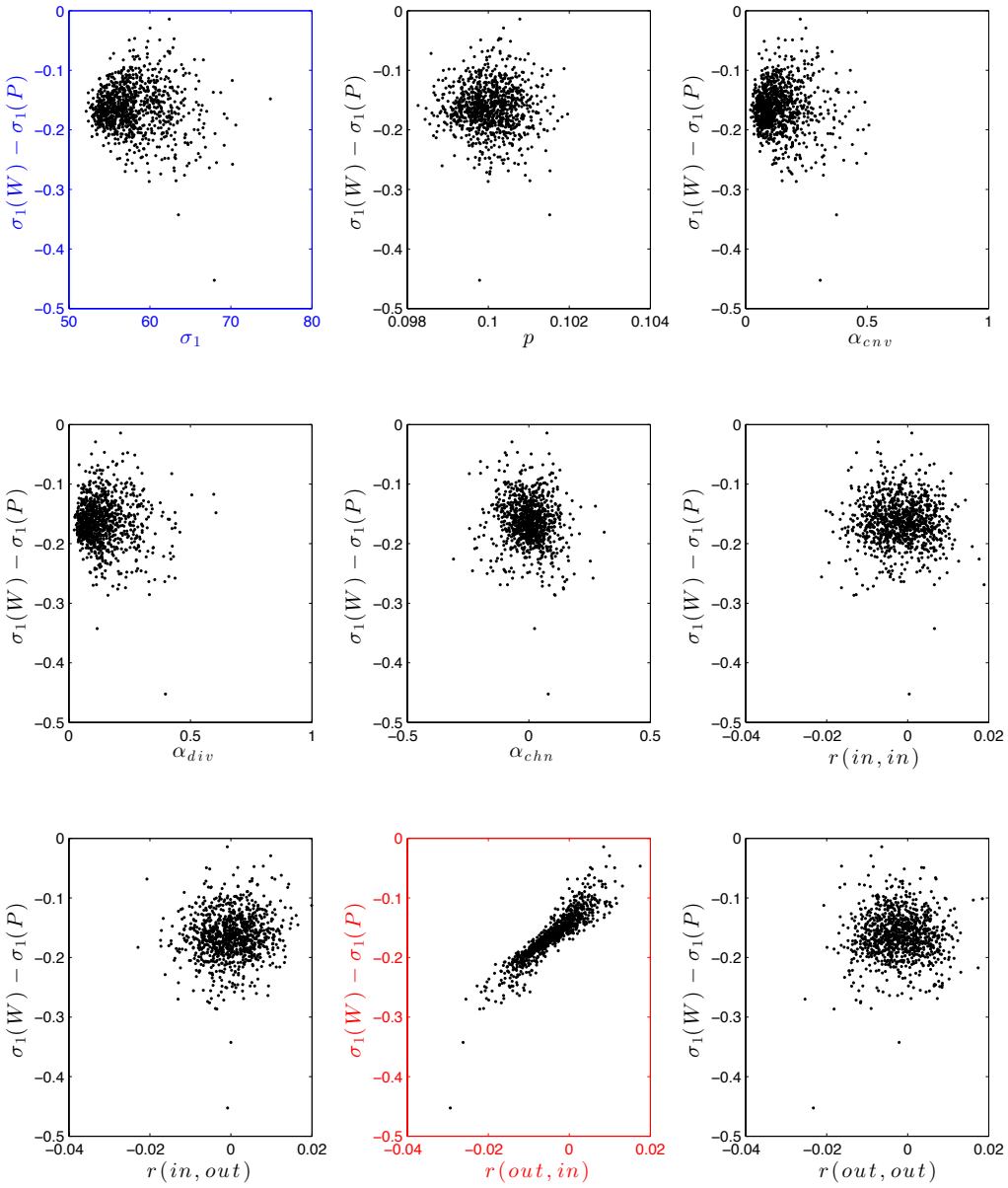


Figure 5.6: **Comparing $\sigma_1(W) - \sigma_1(P)$ with network statistics for the EDM** One thousand networks were generated with $N = 400$, $p \approx 0.1$, and a variety of second order statistics. In every plot, each data point represents one network.The top left (blue frame) plot shows the largest singular value plotted against the difference between the largest singular value and the prediction for the largest singular value using P, $\sigma_1(W) - \sigma_1(P)$. Every other plot has a network statistic along the x-axis and $\sigma_1(W) - \sigma_1(P)$ along the y-axis. Of particular interest is the bottom middle plot (red frame). We can see that even in the case of the EDM the difference $\sigma_1(W) - \sigma_1(P)$ is correlated, and tends to 0 with the value of $r(out, in)$. The other figures are included for comparison.

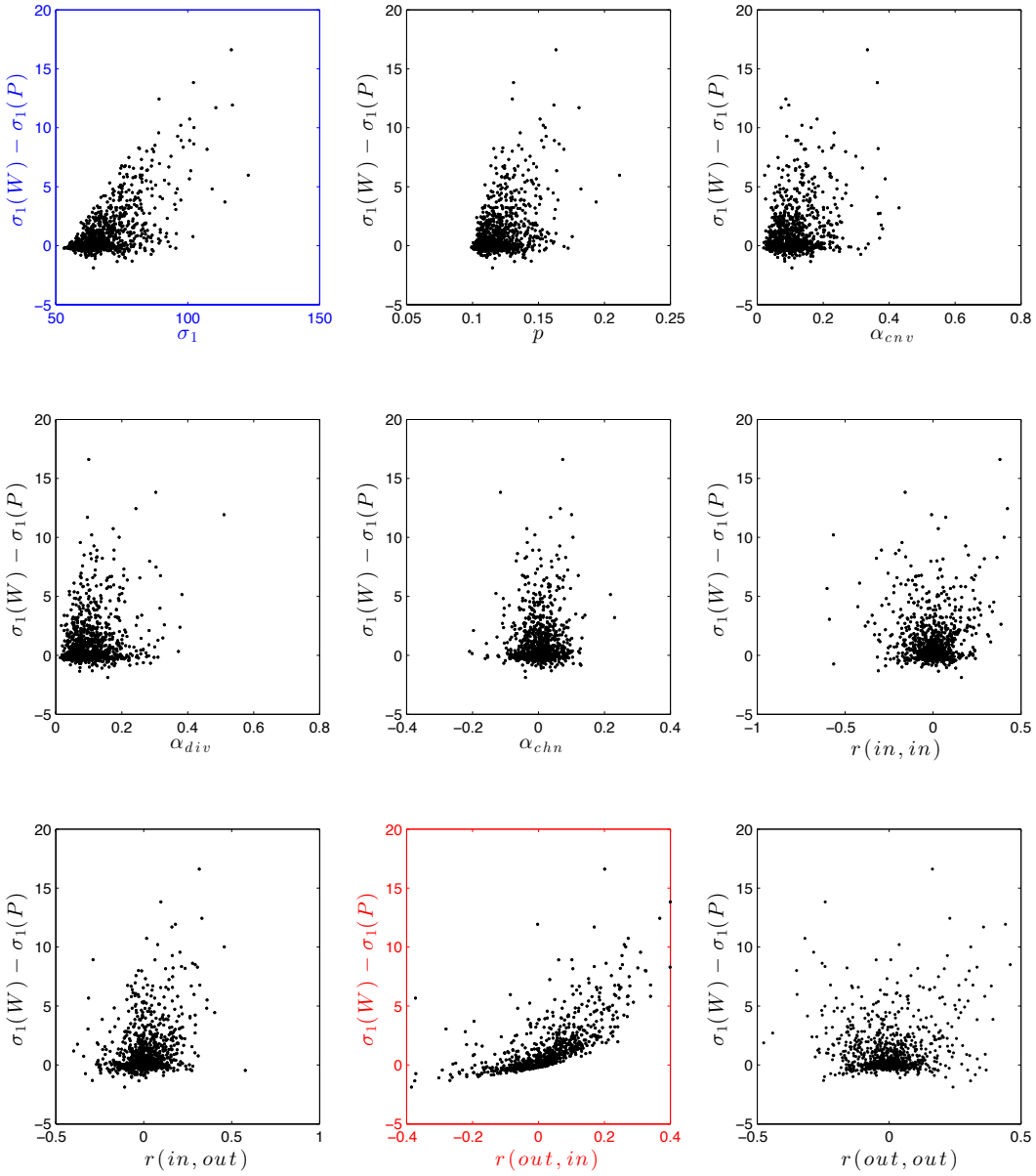


Figure 5.7: **Comparing $\sigma_1(W) - \sigma_1(P)$ with network statistics for the GEDM** One thousand networks were generated with $N = 400$, $p \approx 0.1$, and a variety of second, and third order statistics as in section 3.3.1. In every plot, each data point represents one network.The top left (blue frame) plot shows the largest singular value plotted against the difference between the largest singular value and the prediction for the largest singular value using P, $\sigma_1(W) - \sigma_1(P)$. Every other plot has a network statistic along the x-axis and $\sigma_1(W) - \sigma_1(P)$ along the y-axis. Of particular interest is the bottom middle plot (red frame). Observe that the difference $\sigma_1(W) - \sigma_1(P)$ is most strongly correlated with the value of $r(out, in)$, and tends away from zero with $r(out, in)$. The other figures are included for comparison.

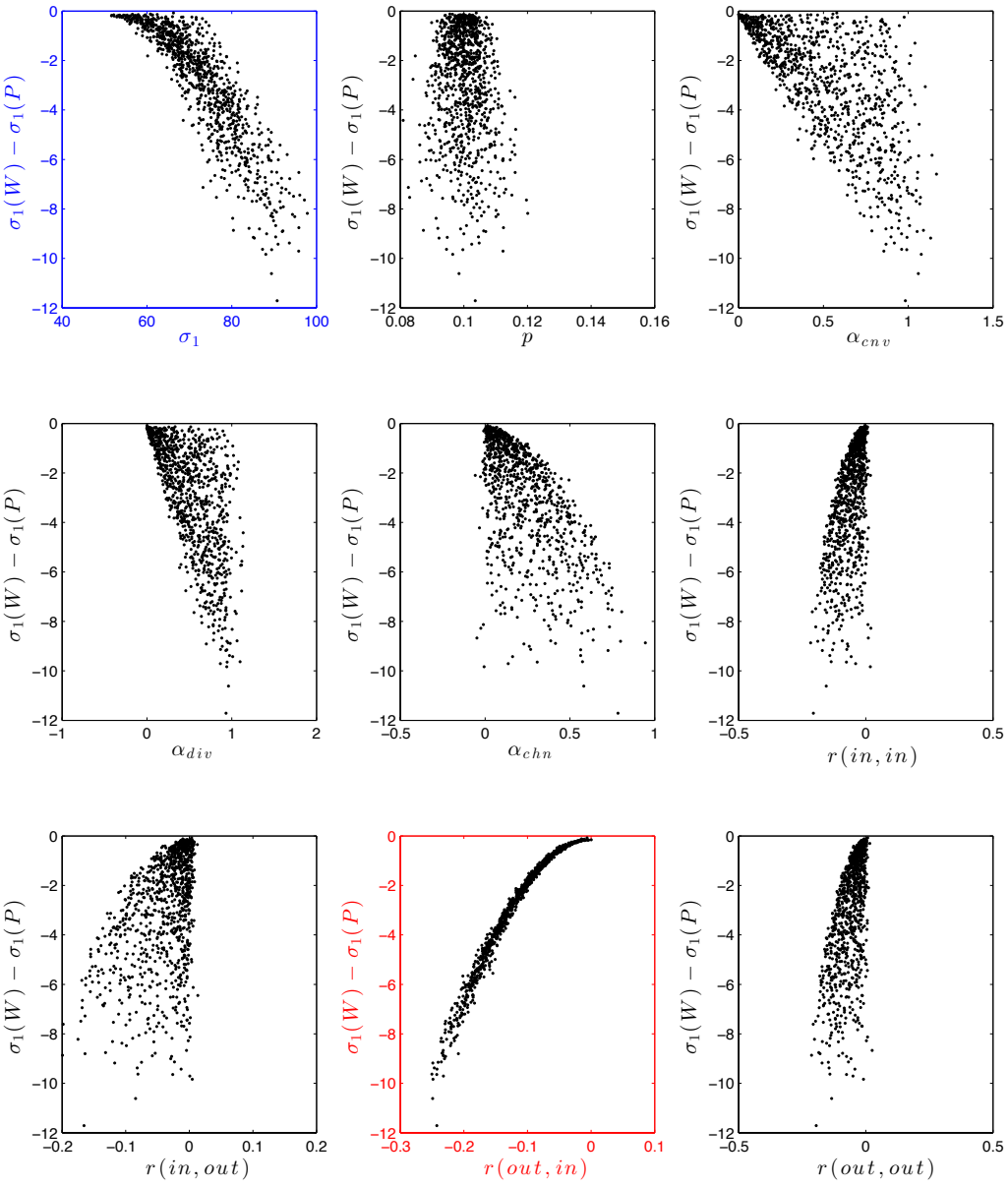


Figure 5.8: **Comparing $\sigma_1(W)-\sigma_1(P)$ with network statistics for the SONET model** One thousand networks were generated with $N = 400$, $p \approx 0.1$, and a variety of second order statistics. In every plot, each data point represents one network. The top left (blue frame) plot shows the largest singular value plotted against the difference between the largest singular value and the prediction for the largest singular value using P, $\sigma_1(W) - \sigma_1(P)$. Every other plot has a network statistic along the x-axis and $\sigma_1(W) - \sigma_1(P)$ along the y-axis. Of particular interest is the bottom middle plot (red frame). The point is that the difference $\sigma_1(W)-\sigma_1(P)$ is most strongly correlated with the value of $r(out, in)$, and tends to 0 with $r(out, in)$. The other figures are included for comparison.

Part II

Dynamics

Chapter 6

The Poisson Spiking Model

I'll now review the *Poisson spiking model* (PSM) that will be the primary dynamical model for the remainder of the thesis. In this chapter I will present the model, and some of its properties. The PSM is remarkable for its simplicity. It may be the simplest spiking model which yields a Wilson-Cowan type rate equation [13], [22]. The PSM is discussed in Gerstner and Kistler's book [22], section 5.2.9 where it is called the *inhomogeneous Poisson model*. Depending on the author, the PSM may be the same or very similar to the Linear-Nonlinear-Poisson model, or the generalized linear model [36].

6.1 Inhomogeneous Poisson processes, spike trains, and shot noise

I'll review the definition and main properties of inhomogeneous Poisson processes (IPP). Details may be found in any text covering elementary stochastic processes (e.g. [39], [43], [13]). Papoulis' book [39] covers much of the material in this section in a fashion which is both elementary and complete.

A *counting process* is a time dependent, nondecreasing random function taking values in the non-negative integers. Generally, A counting process $x(t)$ will have the form

$$x(t) = \sum_{t_k} \Theta(t - t_k),$$

where $\Theta(t)$ is the Heaviside step function

$$\Theta(t) = \begin{cases} 0 & : t \leq 0 \\ 1 & : t > 0. \end{cases} \tag{6.1}$$

The times $\{t_k\}$ will be referred to as the *spike times* of the process, and so our counting processes will be understood to be counting the number of spikes which have occurred up to a given time.

Given a *rate function* $\lambda(t)$, an *inhomogeneous Poisson process* (IPP) with rate $\lambda(t)$ is a counting process $x(t)$ such that the following conditions hold [39].

1. Given times $c < d$, the number $x(d) - x(c)$ is a Poisson random variable with mean
$$\langle x(d) - x(c) \rangle = \int_c^d \lambda(t) \mathrm{d}t.$$
Specifically,
$$P(x(d) - x(c) = k) = \exp\left(- \int_c^d \lambda(t) \mathrm{d}t \right) \frac{1}{k!} \left(\int_c^d \lambda(t) \mathrm{d}t \right)^k.$$

2. Given two intervals $[c, d)$, and $[e, f)$ in $\mathbb{R}$ such that $(c, d) \cap (e, f) = \emptyset$, the random variables $x(f) - x(e)$, and $x(d) - x(c)$ are independent, i.e. the number of spikes which occur one interval do not effect the number of spikes in another independent.

An *(inhomogeneous Poisson) spike train* (IPST) $z(t)$ is a (weak) derivative of an IPP. Thus $z(t)$ will be a sum of delta functions

$$z(t) = \sum_{t_k} \delta(t - t_k),$$

where

$$x(t) = \sum_{t_k} \Theta(t - t_k)$$

is an IPP.

Shot noise is a stochastic process which results from convolving an IPST with a *response function* $h(t)$. Typically one requires (as will we) that $h(t) = 0$, for $t < 0$.

Given spike times $\{t_k\}$, the associated shot noise can be written

$$s(t) = \sum_{t_k} h(t - t_k)$$

The concept of shot noise originated in physics, and has been of interest to physicists for some time [27] having applications to radiative processes.

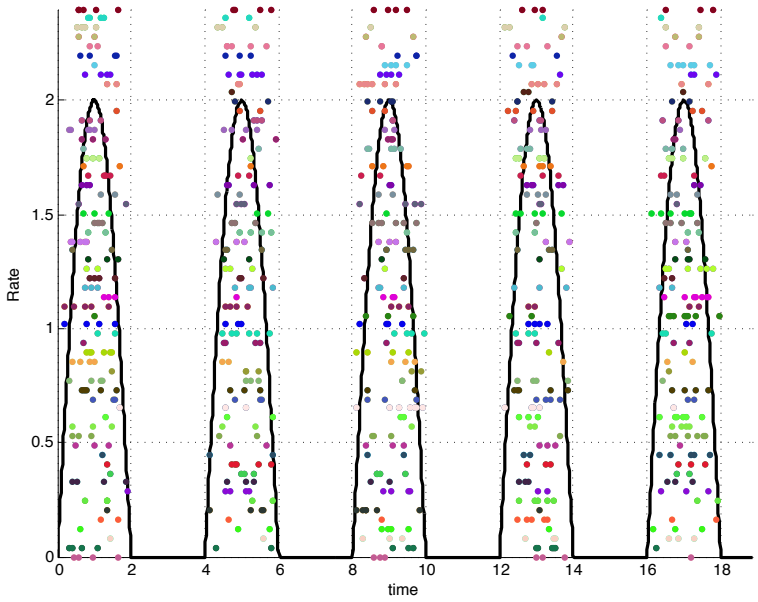


Figure 6.1: **Sixty inhomogeneous Poisson spike trains.** The black curve is the rate function. Spike trains correspond to the rows of dots. Each dot is a spike.

6.2 The Poisson spiking model (PSM), definition and perspectives

The PSM is defined via a response function $h(t)$, a gain function $\phi(g)$ and an adjacency matrix W for a network with N nodes. We associate a function $s^i(t)$ with each node of the network. Given some initial values $s^i(t_0) = s_0^i$, the model evolves stochastically via

$$s^i(t) = s_0^i h(t - t_0) + h * z^i(t), \tag{6.2}$$

where the spike trains,

$$z^i(t) = \sum_{t_k^i} \delta(t - t_k^i) \tag{6.3}$$

have rate functions

$$\lambda^i(t) = \phi\left(\kappa \sum_j W_j^i s^j(t) + b^i(t)\right). \tag{6.4}$$

The scalar κ represents a coupling strength. We will usually consider the case where $b^i(t) = b$ is constant, but we imagine it playing the role of external input to the system.

We note the change of variables:

$$g^i(t) := \kappa \sum_j W_j^i s^j(t) = \kappa \sum_j W_j^i h * z^j(t) = \kappa h * (\sum_j W_j^i z^j(t)). \tag{6.5}$$

Equation (6.5) may be used in lieu of (6.2). The rate functions can be written in terms of the g's directly

$$\lambda^i(t) = \phi(g^i(t) + b^i(t)).$$

We'll call the s's *synapse variables*, and the g's *conductance variables*. See figure 6.2.

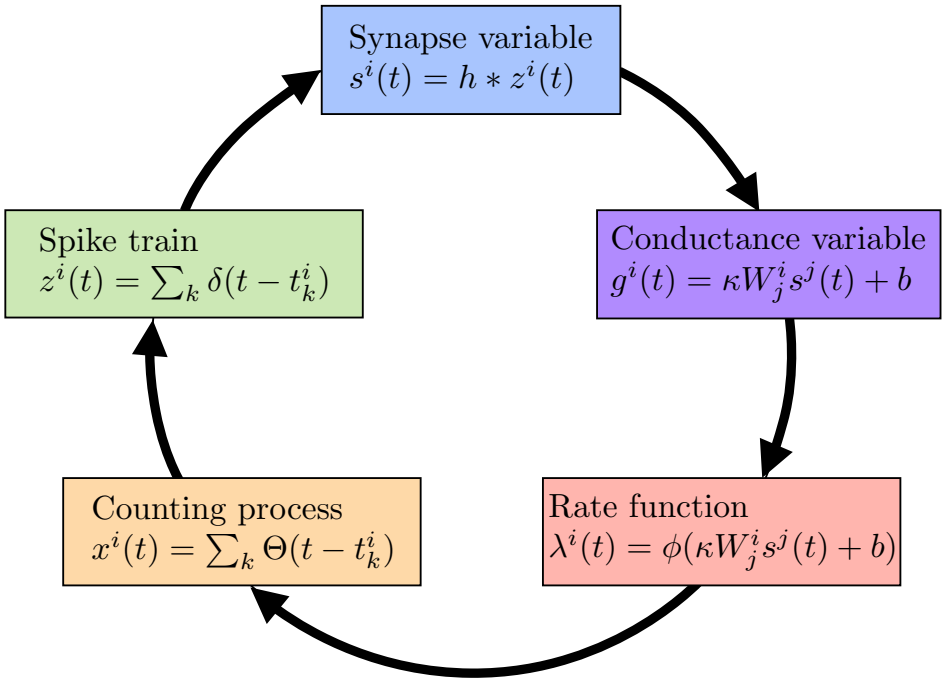


Figure 6.2: **Dependencies of variables for the PSM**

Observe that while one may express the g's in terms of the s's, it is not generally possible to recover the s's given the g's. For example, if $\kappa = 0$, or W is not invertible than s may not be uniquely determined by g. The g's are more sensitive to coupling. Indeed, the s's are only coupled via the *rates* at which the spikes occur, while the g's are coupled directly by via the spikes from other nodes. In particular, if the system is in a saturated regime (for example, if $b^i(t) = b \gg 0$) then s^i and s^j will be independent random processes, while g^i, and g^j will have nontrivial covariance if $W^i_k = W^j_k = 1$ for one or more k.

For the remainder of the thesis I will use the following form of the PSM.

Example 6.2.1. Specifying

$$h(t) = \frac{\alpha}{\tau} \exp\left(-\frac{t}{\tau}\right) \Theta(t), \tag{6.6}$$

the PSM may be written as a first order SDE as

$$\tau \dot{s}^i = -s^i + \alpha \sum_{k=1}^{\infty} \delta(t - t_k^i) \tag{6.7}$$

$$= -s^i + \alpha z^i(t), \tag{6.8}$$

where z^i is a again a spike train with rate function

$$\phi(\kappa \sum_j W_j^i s^j + b^i)$$

If we take the Poisson process as given, the ODE (6.7) is equivalent to the integral equation

$$\begin{aligned} s^i(t) &= (s_0^i + \frac{\alpha}{\tau} \sum_{k=1}^{\infty} \int_0^t e^{-\frac{s}{\tau}} \delta(s - t_k^i) \mathrm{d}s) e^{-t/\tau} \\ &= s_0^i e^{-t/\tau} + \frac{\alpha}{\tau} \sum_{t_k^i < t} e^{-\frac{(t - t_k^i)}{\tau}} \Theta(t - t_k^i) \\ &= s_0^i e^{-t/\tau} + (h * z^i)(t) \end{aligned} \tag{6.9}$$

where s_0^i is the initial value of $s^i(t)$, Θ is the Heaviside function, and the h is given by

Again, taking $g^i = \kappa \sum_j W_j^i s^j$, for each $i = 1, \ldots, N$, we can express the model in terms of the g's. We find

$$\begin{aligned} \tau \dot{g}^i &= -g^i + \alpha \, \kappa W_j^i \sum_{k=1}^{\infty} \delta(t - t_k^j) \\ &= -g^i + \alpha \, \kappa W_j^i \sum_{k=1}^{\infty} z^j(t) \end{aligned} \tag{6.10}$$

where, again, $z^j(t)$ is an IPST with rate $\phi(g^j + b^j)$. The integral form is

$$\begin{aligned} g^i(t) &= g_0^i e^{-t/\tau} + \frac{\alpha\,\kappa}{\tau} W_j^i \sum_{t_k^j < t} e^{-\frac{(t-t_k^j)}{\tau}} \Theta(t - t_k^j) \\ &= g_0^i e^{-t/\tau} + \kappa \sum_j W_j^i (h * z^j)(t) \end{aligned} \tag{6.11}$$

It is worth noting that while the PSM certainly *spikes*, it is *not* generally Poisson. In particular, condition 2 in the definition of an inhomogeneous Poisson process may be explicitly violated, i.e. the number of spikes which occur in a given interval may influence the number of spikes in the next interval. The model does produce an inhomogeneous spike train in two cases: 1) in the absence of input spikes, 2) in a purely feed forward network the spikes of layers receiving spikes are doubly stochastic inhomogeneous poisson processes (aka a Cox process [12]).

Chapter 7

An excursion into path integration

In order to derive some basic estimates for rate equations, and covariances of activity resulting from the Poisson spiking model (PSM) we now take a short detour into path integration. The best references I can give for this section are some articles by Michael Buice, and Carson Chow. The results of this section are not new, and can be found in a forthcoming article from Buice [3]. On the other hand, the arguments in the derivation of the tree level mean and covariance are original. Before applying path integrals to the the PSM, I'll layout some basics of the application of path integrals to stochastic differential equations.

The main references for this chapter are [10], [4]. Also, Chaichian's book [9] is a very nice introduction to the subject of path integration. The application of path integrals to SDE there is in terms of Wiener path integrals, and he summarizes the historical uses of path integrals nicely. The approach that I'll follow in this chapter has its roots in quantum mechanics and quantum field theory originating with Feynman. Feynman has a well written introductory text on path integrals and quantum mechanics [19]. The introductory chapter and chapter 12 are relevant to the application of path integrals to random processes including shot noise. Doi adapted the path integral operator formulation to stochastic processes [15], and Peliti adapted a path integral treatment for birth and death processes on a lattice [40]. For a review of the application of path integrals to simple birth and death

processes I recommend the article by Dickman and Vidigal [14], which may be practically useful.

This chapter is somewhat dense. My main purpose is to derive the approximation to the mean of PSM (7.67), and an approximation to the covariance of the PSM (7.69). In the next chapter I will apply dimension reduction techniques to these equations. Much of chapter 8 is devoted to equations of the form (7.67), and section 8.5 is devoted to development of the tree level covariance (7.69). This material will also serve as background for chapter 9, where path integral methods will be used to derive a Langevin equation corresponding to the PSM.

7.1 Finite dimensional Gaussians

Path integrals are intimately connected practically, and historically with finite dimensional Gaussian distributions [9], [10]. In this section I will briefly review some of the standard results regarding Gaussians. The Gaussian, or normal distribution on $\mathbb{R}^n$ with mean μ, and covariance K (which we assume to be positive definite) has density is

$$\rho(x) = \frac{1}{(2\pi)^{n/2}\det(K)^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^{\mathsf{T}} K^{-1}(x-\mu)\right).$$

Thus given components x_i, x_j we have $\left\langle x^i \right\rangle = \mu^i$, and $\left\langle (x_i - \mu^i)(x_j - \mu^j) \right\rangle = K_{ij}$. The moment generating function associated with ρ is

$$\begin{aligned} Z[J] &= \int \mathrm{d}x\ \rho(x) \exp(x \cdot J) \\ &= \int \mathrm{d}x\ \exp\left(-\frac{1}{2}(x-\mu)^{\mathsf{T}} K^{-1}(x-\mu) + J \cdot x\right) \\ &= \exp\left(J \cdot \mu + \frac{1}{2} J^{\mathsf{T}} K J\right). \end{aligned} \tag{7.1}$$

From the moment generating function one can prove Isserlis' [30] theorem which states that every moment of ρ can be expressed as a sum of all combinations of products of first and second moments. For example,

$$\left\langle x^{i_1} x^{i_2} x^{i_3} \right\rangle = \left\langle x^{i_1} \right\rangle \left\langle x^{i_2} \right\rangle \left\langle x^{i_3} \right\rangle + \left\langle x^{i_1} x^{i_2} \right\rangle \left\langle x^{i_3} \right\rangle + \left\langle x^{i_1} x^{i_3} \right\rangle \left\langle x^{i_2} \right\rangle + \left\langle x^{i_2} x^{i_3} \right\rangle \left\langle x^{i_1} \right\rangle.$$

7.2 Path integrals for stochastic differential equations

Our aim is to write a probability density over the space of sample paths corresponding to a given stochastic differential equation (SDE). In some instances this density may yield an interpretation in terms of Wiener integrals, but for our purposes we will treat them as formal objects. There a number of conveniences that the path integral framework affords. For example, path integrals yield natural ways to organize of the moment hierarchy that may be useful [4]. This section is a brief review of the basic material covered in Buice and Chow's review [10].

7.2.1 The action and path density for SDE

First, let us consider an SDE with Gaussian noise.

$$\mathrm{d}X = F(X,t)\mathrm{d}t + G(X,t)\xi + X_0\delta(t-t_0), \tag{7.2}$$

where $X \in \mathbb{R}^N$, and the noise $\xi \sim \mathcal{N}(0, C(t))$ is delta correlated in time, but may have non-trivial correlations across space. The term with the delta function enforces the initial condition $X(t_0) = X_0$. Let's consider this equation on a time domain $[t_0, T]$. Throughout we interpret SDE's in the Ito sense [21], [18]. As such, one could simulate the process (7.2) by applying the Euler-Maruyama method [32]. That is, we discretize the time interval into $K+1$ steps

$$(t_0, t_1, \ldots, t_K) = (t_0, t_0 + h, \ldots, t_0 + Kh),$$

where $h = (T - t_0)/K$, and iteratively compute the trajectory of the SDE via

$$X_{n+1} = X_n + hF(X_n, nh) + G(X_n, nh)\sqrt{h}\xi_n + X(t_0)\delta_{n,0}, \tag{7.3}$$

where the random variables $\{\xi(t_n)\}$ are distributed according to

$$\xi(t_n) \sim \mathcal{N}(\mu(X(t_{n-1}), t_{n-1}), C(X(t_{n-1}), t_{n-1})).$$

Note, the indices in (7.3) indicate different times. I will use the notation $X_k = X(t_k)$, and $\xi_k = \xi(t_k)$ when there is no ambiguity. In order to apply the path integral

formalism to a stochastic process we only require the process to be Markov, but the increments may not be identically distributed.

For fixed noise terms, the recursion given by (7.3) is deterministic. We can write the distribution over $\mathbb{R}^{K\times N}$ corresponding to the trajectory computed by (7.3) as a product of delta functions:

$$P[(X_1,\ldots,X_K)\,|\,X_0,\,(\xi_1,\ldots,\xi_K)] \\ = \prod_{n=1}^{K}\delta[X_{n+1}-X_n-hF(X_n,nh)-G(X_n,nh)\sqrt{h}\xi_n-X_0\delta_{n,0}]. \tag{7.4}$$

To get an (formal) expression for the density over path space corresponding to (7.2) we want to consider the limit as the step size h goes to zero. Towards that end it is convenient to use the Fourier representation of the Dirac-delta function

$$\delta(x-x_0)=\frac{1}{2\pi}\int_{-\infty}^{\infty}\exp(-i\tilde{x}(x-x_0))\mathrm{d}\tilde{x}.$$

Hence we write

$$P[(X_1,\ldots,X_K)\,|\,X_0,\,(\xi_1,\ldots,\xi_K)] \tag{7.5}$$

$$=\int\left[\prod_{n=1}^{K}\frac{\mathrm{d}\tilde{X}(t_n)}{(2\pi i)^N}\right] \tag{7.6}$$

$$\exp\left(-\sum_{n=1}^{K}\tilde{X}(t_n)(X(t_n)-X(t_{n-1})-hF(X(t_n),t_n)-G(X(t_n),t_n)\sqrt{h}\xi_n-X_0\delta_{n,0})\right),$$

where I have made the substitution $i\tilde{X}\to\tilde{X}$, which explains the factor of i multiplying the 2π. We will call the variable $\tilde{X}$ the *response variable* for X. Others refer to $\tilde{X}$ as *conjugate momenta* [56] hinting at the role of these variables in quantum mechanics. Note: I will assume

X is a column vector and $\tilde{X}$ is row vector.

Therefore, the components of $\tilde{X}$ will be subscripts, as in $\tilde{X}_i$. As such an expression

like $\tilde{X}X$ should be interpreted as

$$\tilde{X}X = \sum_k \tilde{X}_k X^k,$$

an inner product. On the other hand, an expression such as $X\tilde{X}$ would correspond to an outer product.

As an aside, the step of going from the state-space variable X to the phase space pair $(X, \tilde{X})$ puts us in the territory of the Feynman path integral as opposed to the mathematically well defined Wiener path integral. Chaichian [9] does a nice job of comparing and contrasting the two methods.

We now wish to integrate out the noise, ξ_n in (7.5). Up to this point, the only way that the Gaussianity of the noise has impacted our equations is that when numerically integrating (7.2) the noise term gets a factor of $\sqrt{h}$. Other than that, the noise could have been of any type. To *integrate out the noise* means to perform the integral

$$\begin{aligned} &P[(X(t_1), \ldots, X(t_K)) \,|\, X(t_0)] \\ &\quad = \int_{\mathcal{X}} P[(X(t_1), \ldots, X(t_K)) \,|\, X(t_0),\, (\xi(t_1), \ldots, \xi(t_K))] P[(\xi(t_1), \ldots, \xi(t_K))] \mathrm{d}\xi, \end{aligned} \tag{7.7}$$

where $\mathcal{X}$ is whatever space that the noise takes values in. The only term of (7.5) which will effect the integral in (7.7) is

$$\exp\left(\sqrt{h}\sum_{n=1}^{K} \tilde{X}(t_n) G(X(t_n), t_n)\xi(t_n)\right).$$

For simplicity, let $q(t_n) := \sqrt{h}\tilde{X}(t_n)G(X(t_n), t_n)$. Then factoring out terms which don't depend explicitly on ξ, and *assuming* $\xi(t_1), \ldots, \xi(t_K)$ *are independent* the integral in (7.7) takes the form

$$\prod_{n=1}^{K} \int_{\mathcal{X}_n} \exp\left(q(t_n)\xi(t_n)\right) P[\xi(t_n)] \mathrm{d}\xi(t_n). \tag{7.8}$$

Notice that (7.8) is just the product of the moment generating functions associated with $P[\xi(t_n)]$ evaluated at $q(t_n)$. It did not matter that the noise was Gaussian.

Returning to the case of Gaussian noise, using (7.8), and applying (7.1) we arrive at

$$\begin{aligned} &P[(X(t_1),\dots,X(t_K))\,|\,X_0] \\ &= \int \left[\prod_{n=1}^{K} \frac{\mathrm{d}\tilde{X}(t_n)}{(2\pi i)^N}\right] \exp\left(-\sum_{n=1}^{K} \tilde{X}(t_n)(X(t_n)-X(t_{n-1})-hF(X_n,nh)-X_0\delta_{n,0})\right) * \\ &\qquad\qquad \exp\left(\sum_{n=1}^{K} \frac{h}{2}\tilde{X}(t_n)G(X(t_n),t_n)C(t_n)G(X(t_n),t_n)^T\tilde{X}(t_n)^T\right). \end{aligned}$$

Formally, in the limit as $K\to\infty$, $h\to 0$, we have the *path density*

$$\begin{aligned} &\mathcal{P}[X\,|\,X_0] = \\ &\int \mathcal{D}\tilde{X} \exp\left(-\int \mathrm{d}t\{\tilde{X}(t)(\dot{X}(t)-F(X(t),t)-X_0\delta(t-t_0)) \right. \\ &\qquad\qquad \left. -\frac{1}{2}\tilde{X}(t)G(X(t),t)C(t)G(X(t),t)^T\tilde{X}(t)^T\}\right). \end{aligned} \tag{7.9}$$

Note that in (7.9) the normalizing constant (which does not converge as $N\to\infty$) has been absorbed into the symbol $\mathcal{D}$. Generically (7.9) will be $\frac{\infty}{\infty}$, but we will proceed formally. At this point it is beneficial to define the *action* associated with (7.2)

$$\begin{aligned} S[\tilde{X},X] = \int \mathrm{d}t\{&\tilde{X}(t)(\dot{X}(t)-F(X(t),t)-X_0\delta(t-t_0)) \\ &-\frac{1}{2}\tilde{X}(t)G(X(t),t)C(t)G(X(t),t)^T\tilde{X}(t)\}. \end{aligned} \tag{7.10}$$

Thus we may express the path density briefly as

$$\mathcal{P}[X\,|\,X_0] = \int \mathcal{D}\tilde{X} \exp\left(-S[\tilde{X},X]\right). \tag{7.11}$$

For computing statistics associated with the SDE (7.2) it will be convenient to treat the integrand in (7.11) as a path density in the variables $\tilde{X}$, and X jointly. Let

$$\mathcal{P}[\tilde{X},X] = \exp(-S[\tilde{X},X]). \tag{7.12}$$

For our purposes the phrase *path density* will refer to a joint functional of the form

(7.12). One may now proceed to computing moments or cumulants for ensembles of paths for (7.2), but for general actions these computations may be intractable.

7.2.2 Free actions, and the free propagator

One case in which moments at every order can be solved for is when the action is bilinear in $\tilde{X}$ and X. Our approach to approximating moments with respect to the path density (7.12) will be to express the action $S[\tilde{X}, X]$ as a sum of a bilinear part and a nonlinear part, and expand the exponential about the bilinear part. We will then be able to approximate statistics of a general SDE by utilizing the computations for the bilinear part. As such, the computations and definitions in this section will be of paramount importance for estimating statistics of general SDE.

An action $S[\tilde{X}, X]$ is called *free* if it has the form

$$S[\tilde{X}, X] = \int_{t_0}^{\infty} \left(\int_{t_0}^{\infty} \tilde{X}(t) K_F(t, t') X(t') \, \mathrm{d}t' \right) \mathrm{d}t, \tag{7.13}$$

for some operator K_F, i.e. it is bilinear in $\tilde{X}$ and X. The action (7.10) corresponding to the above SDE may fail to be free if

1. F is nonlinear,
2. $X_0 \neq 0$, or
3. $G(X, t)C(t) \neq 0$ for some t (assuming continuity of G, and C).

If we had started with the linear ordinary differential equation

$$\dot{X} = AX, \tag{7.14}$$

with $X(t_0) = 0$ then the corresponding action would be free (and the problem would be trivial). In this case we would have

$$K_F(t, t') = \delta(t - t') \left(\frac{\mathrm{d}}{\mathrm{d}t} - A \right). \tag{7.15}$$

Notice, we can express (7.14) as

$$\int_{t_0}^{T} K_F(t,t')X(t')\mathrm{d}t' = 0. \tag{7.16}$$

If we were to impose an arbitrary initial condition $X(t_0) = X_0$, we could write the differential equation as

$$\int_{t_0}^{T} K_F(t,t')X(t')\mathrm{d}t' = X_0\delta(t-t_0). \tag{7.17}$$

The *Green's function*, or *free propagator*, corresponding to (7.17) is an operator $\Delta_F(t,t')$ such that

$$\int_{-\infty}^{\infty} \mathrm{d}t' K_F(t,t')\Delta_F(t',t'') = \int_{-\infty}^{\infty} \mathrm{d}t' \Delta_F(t,t')K_F(t',t'') = I_N\delta(t-t''), \tag{7.18}$$

where I_N is the $N \times N$ identity matrix. The free propagator associated with K_F as in (7.15), is simply

$$\Delta_F(t,t') = H(t-t')\exp\left(A(t-t')\right), \tag{7.19}$$

where $H(t)$ is the Heaviside function. Throughout, I will follow the convention that $H(x) = 0$ for $x \leq 0$, and $H(x) = 1$ for $x > 0$. The reason for the name *propagator* stems from the fact that we can express any solution of (7.17) as

$$X(t) = \int_{t_0}^{\infty} \mathrm{d}t' \Delta_F(t,t')X_0\delta(t'-t_0) = \Delta(t,t_0)X_0.$$

In light of the identity (7.18), the kernel K_F of the free action (7.15) is called the *free inverse propagator*. With this in mind I will change notation. Henceforth, the free inverse propagator will be written as

$$\Delta_F^{-1}(t,t') := K_F(t,t'). \tag{7.20}$$

This notation emphasizes the key role that the propagator plays in the path integral framework, as will be made more clear in the next section.

7.2.3 Computing moments with respect to a free path density

By a *free path density* or simply *free density* I mean a path density as in (7.12) for which the action $S[\tilde{X}, X]$ is free (i.e. bilinear in $\tilde{X}$, and X). We will now consider the moments of $\tilde{X}$, and X with respect to a *free density*

$$\mathcal{P}[\tilde{X}, X] = \exp\left(-\sum_{i,j}\int_{t_0}^{\infty}\int_{t_0}^{\infty}\tilde{X}_i(t)\Delta_F^{-1}(t,t')^i_j X^j(t')\mathrm{d}t\mathrm{d}t\right). \tag{7.21}$$

These moments can be computed by taking functional derivatives of the *free moment generating functional*

$$\begin{aligned} Z_F[J,\tilde{J}] &= \int \mathcal{D}\tilde{X}\mathcal{D}X \exp\left(-\tilde{X}(t)\cdot\Delta_F^{-1}(t,t')\cdot X(t') + \tilde{X}\cdot J + \tilde{J}\cdot X\right) \\ &=: \int \mathcal{D}\tilde{X}\mathcal{D}X \exp\left(\mathcal{I}\right), \end{aligned} \tag{7.22}$$

where

$$\tilde{f}(t)\cdot g(t) = \tilde{f}\cdot g = \sum_k \int \mathrm{d}t \tilde{f}_k(t) g^k(t).$$

Note the convention: when taking the inner product of two functions the integral is taken over their common argument, if one is provided. For example, in (7.22) we have

$$\tilde{X}(t)\cdot\Delta_F^{-1}(t,t') = \int \mathrm{d}t \tilde{X}(t)\Delta_F^{-1}(t,t').$$

This convention will greatly reduce the length of the expressions below.

We can "compute" the path integral (7.22) by performing the change of variables

$$\begin{aligned} Y(t) &= X(t) - \Delta_F(t,t')\cdot J(t') \\ \tilde{Y}(t') &= \tilde{X}(t') - \tilde{J}_l(t)\cdot\Delta_F(t,t'). \end{aligned} \tag{7.23}$$

If $\mathcal{D}X$ behaves like a usual measure, then we can assume $\mathcal{D}Y = \mathcal{D}X$ and $\mathcal{D}\tilde{Y} = \mathcal{D}\tilde{X}$.

Applying the change of variable (7.23) to the first term of $\mathcal{I}$ in (7.22) yields

$$\begin{aligned}
&- \tilde{X}(t) \cdot \Delta_F^{-1}(t,t') \cdot X(t') \\
&= -\left(\tilde{Y}(t) + \tilde{J}(t'') \cdot \Delta_F(t'',t)\right) \cdot \Delta_F^{-1}(t,t') \cdot \left(Y(t') + \Delta_F(t',t''') \cdot J^l(t''')\right) \\
&= -\tilde{Y}(t) \cdot \Delta_F^{-1}(t,t') \cdot Y^j(t') - \tilde{J} \cdot Y - \tilde{Y} \cdot J - \tilde{J}(t) \cdot \Delta_F(t,t') \cdot J(t').
\end{aligned} \tag{7.24}$$

The second, and third terms of $\mathcal{I}$ become

$$\tilde{Y} \cdot J + \tilde{J}(t) \cdot \Delta_F(t,t') \cdot J(t'), \tag{7.25}$$

and

$$\tilde{J} \cdot Y + \tilde{J}(t) \cdot \Delta_F(t,t') \cdot J(t'). \tag{7.26}$$

Combining (7.24), (7.25), and (7.26) we find

$$\begin{aligned}
Z_F[\tilde{J},J] &= \int \mathcal{D}\tilde{X}\mathcal{D}X \exp(-S_F[\tilde{X},X] + \tilde{J} \cdot X + \tilde{X} \cdot J) \\
&= \int \mathcal{D}\tilde{Y}\mathcal{D}Y \exp(-S_F[\tilde{Y},Y]) \exp\left(\tilde{J}(t) \cdot \Delta_F(t,t') \cdot J(t')\right) \\
&= \exp\left(\tilde{J}(t) \cdot \Delta_F(t,t') \cdot J(t')\right).
\end{aligned} \tag{7.27}$$

As mentioned above the moments of $\mathcal{P}[\tilde{X},X]$ are computed by taking functional derivatives of the moment generating function, and setting $\tilde{J} = J = 0$. From (7.27), the moment generating function for a free action one can prove Wick's theorem, which is analogous to Isserlis' theorem for Gaussian integrals.

Theorem 7.2.1. [10]

1. Given indices i, and j we have $\left\langle X^i(t)\tilde{X}_j(t')\right\rangle_F = \Delta_F(t,t')^i_j$, where

$$\langle\cdot\rangle_F \tag{7.28}$$

   denotes the moment with respect to a free action, and

2. the only non vanishing moments of a free path density are those with an equal number of $\tilde{X}$ and X components. Any such moment can be expressed as a sum over all pairings of $\tilde{X}$ components with X components of the moments of

the pairs.

To illustrate the second part of this theorem, we have

$$\left\langle X^i(t)X^j(t')\tilde{X}_k(t'')\tilde{X}_l(t''')\right\rangle_F$$

$$= \frac{\delta}{\delta\tilde{J}_i(t)}\frac{\delta}{\delta\tilde{J}_j(t')}\frac{\delta}{\delta J^k(t'')}\frac{\delta}{\delta J^l(t''')}\bigg|_{\tilde{J}=J=0} Z_F[\tilde{J},J] \tag{7.29}$$

$$= \left\langle X^i(t)\tilde{X}_k(t'')\right\rangle_F\left\langle X^j(t')\tilde{X}_l(t''')\right\rangle_F + \left\langle X^i(t)\tilde{X}_l(t''')\right\rangle_F\left\langle X^j(t')\tilde{X}_k(t'')\right\rangle_F$$
$$= \Delta_F(t,t'')^i_k\Delta_F(t',t''')^j_l + \Delta_F(t,t''')^i_l\Delta_F(t',t'')^j_k.$$

In the next section I will show how to apply theorem 7.2.1 to compute moments of a linear Ornstein-Uhlenbeck process.

7.2.4 Computing the moments for the linear Ornstein-Uhlenbeck process

I will now show how to apply the path integral framework introduced above to compute the first two moments of a multivariate linear Ornstein-Uhlenbeck (OU) process. In particular, we will see how to apply theorem 7.2.1 to compute moments with respect to a path density which has an action that isn't free. The results of this section (the mean and covariance for the OU process) are well known, but are included here to illustrate, and further develop how one may compute moments for an SDE within the path integral framework.

Consider the SDE over $\mathbb{R}^N$

$$\mathrm{d}X = AX\mathrm{d}t + B\mathrm{d}\xi + X_0\delta(t-t_0), \tag{7.30}$$

where $\mathrm{d}\xi$ is M-dimensional white noise, A is an $N\times N$ matrix and B is a $N\times M$ matrix. We want to compute the mean and covariance of this system. The action

corresponding to (7.30) can be written using (7.10)

$$S[\tilde{X}, X] = \int_{t_0}^{\infty} \mathrm{d}t \left(\tilde{X}(t)(\dot{X}(t) - AX(t) - X_0\delta(t - t_0)) - \frac{1}{2}\tilde{X}(t)BB^T\tilde{X}(t)^T \right). \tag{7.31}$$

Note that X_0 is a constant vector indicating initial conditions and is *not* treated as $X(0)$. Rather, the fact that $X(0) = X_0$ is a consequence of the delta function. As before, the (full) path density associated with the action (7.31) is simply

$$\mathcal{P}[\tilde{X}, X] = \exp\left(-S(\tilde{X}, X]\right). \tag{7.32}$$

I will denote expectations with respect to the full path density by $\langle\cdot\rangle$.

Remark 7.2.2. To compute moments of the system our first step is to express the action as a sum of a *free part* which consists of the part of the action which is bilinear in $\tilde{X}$ and X, and an *interacting part* consisting of the remaining part of the action. To clarify, let $\mathcal{S}[\tilde{X}, X]$ be an arbitrary action, and let $\tilde{Y}$, and Y be two fixed paths. We will expand the action $\mathcal{S}$ about the paths $\tilde{Y}$, and Y. Expanding $\mathcal{S}$ (in terms of functional derivatives, see 7.4) about $\tilde{Y}$, and Y yields

$$\begin{aligned}\mathcal{S}[\tilde{X}, X] = \mathcal{S}[\tilde{Y}, Y] + \left(\tilde{X}(t) - \tilde{Y}(t)\right) \cdot \frac{\delta}{\delta\tilde{X}(t)}\mathcal{S}[\tilde{Y}, Y] + \frac{\delta}{\delta X(t)}\mathcal{S}[\tilde{Y}, Y] \cdot (X(t) - Y(t)) \\ + \frac{1}{2}\left(\tilde{X}(t) - \tilde{Y}(t)\right) \cdot \frac{\delta^2}{\delta\tilde{X}(t)\delta X(t')}\mathcal{S}[\tilde{Y}, Y] \cdot \left(X(t') - Y(t')\right) + \dots.\end{aligned} \tag{7.33}$$

The *free part of the action with respect to* $\tilde{Y}$ *and* Y is simply

$$\mathcal{S}_F[\tilde{X}, X] = \frac{1}{2}\left(\tilde{X}(t) - \tilde{Y}(t)\right) \cdot \frac{\delta^2}{\delta\tilde{X}(t)\delta X(t')}\mathcal{S}[\tilde{Y}, Y] \cdot \left(X(t') - Y(t')\right), \tag{7.34}$$

and the *interacting part with respect to* $\tilde{Y}$ *and* Y is the remainder:

$$\mathcal{S}_I[\tilde{X}, X] = \mathcal{S}[\tilde{X}, X] - \mathcal{S}_F[\tilde{X}, X]. \tag{7.35}$$

As I have defined it, the free part of the action with respect to $\tilde{Y}$, and Y will always be bilinear in $\tilde{X}(t) - \tilde{Y}(t)$, and $X(t') - Y(t')$.

As we have seen above (7.20), the kernel of the free part of the action is referred

to as the *free inverse propagator*. That terminology, though, customarily refers to the kernel of the free part of the action resulting from the choice $\tilde{Y} = Y = 0$. On the other hand, the kernel of the free part of the action with respect to $\tilde{Y} = 0$, and $Y \approx \langle X \rangle$, is referred to as the *inverse tree level propagator*. The meaning of the approximation $Y \approx \langle X \rangle$, and the concept of *tree level* will be clarified in section 7.3 where we will approximate the mean and covariance for the Poisson spiking model to tree level.

Returning to the linear OU process, expanding the action about the $\tilde{X} = X = 0$, yields the free part of the action

$$S_F[\tilde{X}, X] = \int_{t_0}^{\infty} \mathrm{d}s \ \mathrm{d}t \ \tilde{X}(s) \left(\frac{\mathrm{d}}{\mathrm{d}t} - A \right) X(t) \delta(s - t), \tag{7.36}$$

and the interacting part is

$$S_I[\tilde{X}, X] = -\int_{t_0}^{\infty} \mathrm{d}t \tilde{X}(t) \delta(t - t_0) X_0 - \frac{1}{2} \int_{t_0}^{\infty} \mathrm{d}t \tilde{X}(t) B B^T \tilde{X}(t)^T. \tag{7.37}$$

To compute moments of the SDE (7.30) we start by expanding the path density (7.32) around the free part of the action

$$\begin{aligned} \mathcal{P}[\tilde{X}, X] &= \exp\left(-S[\tilde{X}, X] \right) \\ &= \exp\left(-S_F[\tilde{X}, X] - S_I[\tilde{X}, X] \right) \\ &= \exp\left(-S_F[\tilde{X}, X] \right) \left(\sum_{n=0}^{\infty} \frac{(-S_I[\tilde{X}, X])^n}{n!} \right) \\ &= \sum_{n=0}^{\infty} \frac{(-S_I[\tilde{X}, X])^n}{n!} \exp\left(-S_F[\tilde{X}, X] \right). \end{aligned} \tag{7.38}$$

We may now compute moments with respect to the path density by applying Wick's theorem 7.2.1 for the free density to each of the terms in the series[1]. For example,

[1]Depending on the form of $S_I[\tilde{X}, X]$, the number of terms in the series expansion in (7.38) can grow very rapidly with n. *Feynman diagrams* [10] are graphical representations for terms like those in the expansion above (7.38), and are useful in a number of ways (e.g. eliminating terms, identifying recursions, and organizing the hierarchy of moments).

to compute the mean value of the system we need to calculate

$$\begin{aligned}\langle X(t)\rangle &= \int \mathcal{D}\tilde{X}\mathcal{D}X\ X(t)\exp(-S[\tilde{X},X]) \\ &= \sum_{n=0}^{\infty}\frac{1}{n!}\int \mathcal{D}\tilde{X}\mathcal{D}X\ X(t)(-S_I[\tilde{X},X])^n\exp(-S_F[\tilde{X},X]) \\ &= \sum_{n=0}^{\infty}\frac{1}{n!}\left\langle X(t)(-S_I[\tilde{X},X])^n\right\rangle_F \\ &= \sum_{n=0}^{\infty}\frac{1}{n!}\left\langle X(t)\left(\int_{t_0}^{\infty}\mathrm{d}t'\tilde{X}(t')\delta(t'-t_0)X_0+\frac{1}{2}\int_{t_0}^{\infty}\mathrm{d}t'\,\tilde{X}(t')BB^T\tilde{X}(t')^T\right)^n\right\rangle_F .\end{aligned} \tag{7.39}$$

By Wick's theorem the only nonzero terms of (7.39) will be those which have an equal number of X's and $\tilde{X}$'s. For this system, that means only the initial condition term when $n=1$ will effect the mean. We find

$$\langle X(t)\rangle = \left\langle X(t)\tilde{X}(t_0)\right\rangle_F X_0 = \Delta_F(t,t_0)X_0. \tag{7.40}$$

Similarly for the second moment we find

$$\begin{aligned}&\left\langle X^i(t)X^j(t')\right\rangle \\ &=\sum_{n=0}^{\infty}\frac{1}{n!}\left\langle X^i(t)X^j(t')\left(\int_{t_0}^{\infty}\mathrm{d}t''\tilde{X}(t'')\delta(t''-t_0)X_0+\frac{1}{2}\int_{t_0}^{\infty}\mathrm{d}t''\ \tilde{X}(t'')BB^T\tilde{X}(t'')^T\right)^n\right\rangle_F .\end{aligned} \tag{7.41}$$

We will only have an equal number of X and $\tilde{X}$ components in this expression when $n=1$, and $n=2$. It will be clearer to proceed in coordinates. When $n=1$ we get the term

$$\begin{aligned}&\frac{1}{2}\int_{t_0}^{\infty}\mathrm{d}t''\sum_{k,l,m}\left\langle X^i(t)X^j(t')\tilde{X}_k(t'')\tilde{X}_l(t'')\right\rangle_F B^k_m B^l_m \\ &=\frac{1}{2}\int_{t_0}^{\infty}\mathrm{d}t''\sum_{k,l,m}\left(\Delta_F(t,t'')^i_k\Delta_F(t',t'')^j_l+\Delta_F(t,t'')^i_l\Delta_F(t',t'')^j_k\right)B^k_m B^l_m \\ &=\sum_{k,l,m}\int_{t_0}^{\infty}\mathrm{d}t''\Delta_F(t,t'')^i_k\Delta_F(t',t'')^j_l B^k_m B^l_m .\end{aligned} \tag{7.42}$$

When $n=2$ we also get a contribution from the initial condition term which amounts

to the product of the means. Therefore the full second moment is

$$\begin{aligned}&\left\langle X^i(t)X^j(t')\right\rangle \\ &= \sum_{k,l,m}\int_{t_0}^{\infty} \mathrm{d}t''\Delta_F(t,t'')^i_k\Delta_F(t',t'')^j_l B^k_m B^l_m + \sum_{k,l}\Delta_F(t,t_0)^i_k X^k_0 \Delta_F(t',t_0)^j_l X^l_0.\end{aligned} \tag{7.43}$$

Considering the expression we got for the mean (7.39), we find that (7.43) can be expressed as

$$\left\langle X^i(t)X^j(t')\right\rangle = \sum_{k,l,m}\int_{t_0}^{\infty} \mathrm{d}t''\Delta_F(t,t'')^i_k\Delta_F(t',t'')^j_l B^k_m B^l_m + \left\langle X^i(t)\right\rangle \left\langle X^j(t')\right\rangle. \tag{7.44}$$

To complete the calculation we need to solve for the free propagator. The free inverse propagator is the kernel of the free part of the action (7.36)

$$\Delta_F^{-1}(t,t') = \delta(t-t')\left(\frac{\mathrm{d}}{\mathrm{d}t} - A\right).$$

(Notice that this is the same as the free inverse propagator for the linear ODE (7.15).) Again, the free propagator satisfies

$$\int \mathrm{d}t'\Delta_F^{-1}(t,t')\Delta_F(t',t'') = I_N\delta(t-t''), \tag{7.45}$$

where I_N is the $N \times N$ identity matrix. Equation (7.45) implies

$$\Delta_F(t,t') = H(t-t')\exp\left(-A(t-t')\right). \tag{7.46}$$

where H is the Heaviside function. The convention that $H(0) = 0$ is consistent with the Ito interpretation of the SDE [10].

In summary we have found that mean and covariances for the linear OU process (7.30) satisfy

$$\left\langle X^i(t)\right\rangle = H(t-t_0)\exp(-A(t-t_0))^i_j X^j_0, \tag{7.47}$$

and

$$\left\langle X^i(t)X^j(t')\right\rangle_c = \sum_{k,l,m}\int_{t_0}^{\min(t,t')} \mathrm{d}t''\exp(-A(t-t''))^i_k \exp(-A(t'-t''))^j_l B^k_m B^l_m. \tag{7.48}$$

Throughout this thesis, the sub-c on the angled brackets refers to the cumulant [39], which is the same as covariance at second order, i.e.

$$\left\langle X^i(t)X^j(t')\right\rangle_c := \left\langle X^i(t)X^j(t')\right\rangle - \left\langle X^i(t)\right\rangle\left\langle X^j(t')\right\rangle. \tag{7.49}$$

The results (7.47), and (7.48) are well known, and can be computed by other methods (for example, see [21]). The above derivation is intended to be instructive, as it is simple enough to compute the moments in full, and see some of the ways that one can apply the path integral machinery.

7.3 Path integrals for the Poisson spiking model

We now return to the Poisson spiking model (PSM). I'd like to reemphasize that these results were previously derived by Michael Buice in a forthcoming article [3], but that the derivation below is original work. Using essentially the same steps as in section 7.2.1 above we can derive an action for the PSM. After exhibiting the action I'll describe how one can go about computing basic estimates for the mean and covariance for the PSM.

Suppose we have a N nodes connected by the adjacency matrix W. Recall, the dynamics of the PSM are given, as in example 6.2.1, by

$$\tau\dot{s}^i = -s^i + \sum_k \delta(t-t^i_k), \tag{7.50}$$

where the spike times $\{t^i_k\}$ of node i are produced by an inhomogeneous Poisson process with instantaneous rate $\phi(\kappa W^i_j s^j(t) + b^j)$. The function ϕ is typically taken to be sigmoidal.

To derive the action one proceeds as above starting with the Euler-Maruyama

iteration

$$s^i(t_{n+1}) = s^i(t_n + h) = s^i(t_n) - \frac{h}{\tau}s^i(t_n) + \frac{1}{\tau}\eta_n^i + s(t_0)\delta_{n,0}, \tag{7.51}$$

where $\eta_n^i \sim \text{Poiss}\left(h\phi\left(\kappa W_j^i s^j(t_n) + b^j\right)\right)$. One may now follow exactly the same steps as for the action in the case of Gaussian noise. There is one issue with this however, and that is that η_n^i and η_m^j may not be independent. In particular, if there is an edge from j to i, then a spike at j will change the probability of a spike at i. I will treat them as independent however, and justify this by assuming the coupling is weak (e.g. $\kappa \approx 1/N$, with N large), and so the trial to trial fluctuations in the noise have little direct effect in the firing rates. In other words, the behavior of the system is dominated by the average activity, or at least statistics of the activity, and not on the specific spike times. This limits the applicability of the model. If the rate equations (mean field theory) are near a bifurcation where a spike could trigger a qualitative change in dynamics we do not expect our equations to hold.

The moment generating function for a Poisson variable with mean r is

$$Z[J] = \exp((e^J - 1)r). \tag{7.52}$$

Applying (7.52) and following the same steps as in the Gaussian noise case yields the following action for the PSM:

$$\begin{aligned} S[\tilde{s}, s] = \sum_i \int_{t_0}^{\infty} \mathrm{d}t\ \tilde{s}_i(t)\left(\dot{s}^i(t) + \frac{s^i(t)}{\tau}\right) \\ - \left(\exp\left(\frac{\tilde{s}_i(t)}{\tau}\right) - 1\right)\phi(\kappa W_j^i s^j(t) + b^i) - \tilde{s}_i(t)\delta(t - t_0)s_0^i. \end{aligned} \tag{7.53}$$

The path density for the PSM is

$$\mathcal{P}[\tilde{s}, s] = \exp\left(-S[\tilde{s}, s]\right) \tag{7.54}$$

As indicated in remark 7.2.2, we will approximate moments with respect to the density (7.54) to *tree level*. Note, this derivation is my own work, and differs from the one in Buice's paper [3]. The notation and results will be consistent with those in [3], however. The name "tree level" refers to the fact that in this approximation

we only include Feynman diagrams which are trees [10]. In terms of the action, it means that we will only consider the nonlinearity, ϕ, to first order. The result is that we will neglect the effects that moments of s of second degree or higher might have on the mean. For the remainder of the chapter, I will use $\langle\cdot\rangle_T$ to denote the tree level expectation. The computation will proceed as follows:

1. Expand ϕ around a presently unspecified function $s = \bar{s}^i$ (which later we will set to be the tree level mean itself) to first order.

2. Perform a change of variable $s^i \to \delta s^i := s^i - \bar{s}^i$ in the action and recollect terms. The resulting action $S_{eff}[\tilde{s}, \delta s]$ will be referred to as the *effective action*. The path density associated with the effective action, captures the statistics of fluctuations of s about the tree level mean $\bar{s}$ ignoring effects of correlations between the s variables on the firing rates $\phi(\kappa W^i_j s^j + b^i)$. The free part of the effective action (that part bilinear in $\tilde{s}$ and δs) yields the *inverse tree level propagator*.

3. Determine the value of $\bar{s}$ self consistently by setting $\langle \delta s \rangle_T = 0$. In other words, we set $\bar{s} = \langle s \rangle_T$.

4. Compute the tree level covariance of s by expanding the effective action about the bilinear part (the "free" part, having kernel equal to the inverse tree level propagator), and applying Wick's theorem exactly as was done for the OU process in section 7.2.4.

7.3.1 Computing the effective action, and tree level propagator for the PSM

In this section, I will complete the first two steps above. In order to mitigate the number of symbols define

$$\psi^i(t) := \phi(\kappa W^i_j \bar{s}^j(t) + b^i),$$

and

$$d\psi^i(t) := \phi'(\kappa W^i_j \bar{s}^j(t) + b^i).$$

Note, that ψ, and $d\psi$ are functions evaluated at the auxiliary function $\bar{s}$. Expanding ϕ to first order around $\bar{s}$, and using the ψ's we have

$$\phi\left(\kappa W_j^i s^j(t) + b^i\right) \approx \psi^i(t) + \kappa\, d\psi^i(t)\, W_k^i\, \delta s^k(t). \tag{7.55}$$

Plugging (7.55) into the action and collecting terms yields the *effective action*

$$\begin{aligned} S_{eff}[\tilde{s}, \delta s] = \sum_{i,j} \int_{t_0}^{\infty} \mathrm{d}t \int_{t_0}^{\infty} \mathrm{d}t' \tilde{s}_i(t) \Delta_T^{-1}(t,t')_j^i \delta s^j(t') \\ + \sum_i \int_{t_0}^{\infty} \tilde{s}_i(t) \left(\dot{\bar{s}}^i(t) + \frac{1}{\tau}\left(\bar{s}(t)^i - \psi^i(t)\right) - s_0^i \delta(t - t_0) \right) \\ - \sum_i \int_{t_0}^{\infty} \left(\sum_{n=2}^{\infty} \frac{1}{n!} \left(\frac{\tilde{s}_i(t)}{\tau} \right)^n \right) \left(\psi^i(t) + \kappa\, d\psi^i(t)\, W_j^i\, \delta s^j(t) \right). \end{aligned} \tag{7.56}$$

The *inverse tree level propagator* Δ_T^{-1}, appearing in (7.56) is

$$\Delta_T^{-1}(t,t')_j^i = \delta(t-t') \left(\left(\frac{\mathrm{d}}{\mathrm{d}t} + \frac{1}{\tau} \right) \delta_j^i - \frac{\kappa}{\tau}\, d\psi^i(t)\, W_j^i \right). \tag{7.57}$$

We will use the effective action to compute the tree level statistics. The reason we don't call this the tree level action is because this could lead to confusion since the inverse tree level propagator only appears in the bilinear (free) part of the effective action.

Define

$$\Gamma_j^i(t) = \frac{1}{\tau}\left(\delta_j^i + \kappa\, d\psi^i(t)\, W_j^i\right). \tag{7.58}$$

As with the free propagator in (7.18), the *tree level propagator* satisfies

$$\int_{t_0}^{\infty} \mathrm{d}t' \Delta_T^{-1}(t,t') \Delta_T(t',t'') = I_N \delta(t - t''). \tag{7.59}$$

Applying the definition of Δ_T^{-1} in (7.57), and using Γ we find (7.59) is equivalent to

$$\frac{\mathrm{d}}{\mathrm{d}t} \Delta_T(t,t')_j^i = -\Gamma_k^i(t) \Delta_T(t,t')_j^k + \delta_j^i \delta(t - t'). \tag{7.60}$$

7.3.2 Computing $\langle s\rangle_T$, and $\langle s(t)s(t')\rangle_T$

To compute the mean to tree level, we're first going to compute $\langle \delta s^i(t)\rangle_T$ with respect to the density with action (7.56). The tree level mean $\langle s\rangle_T$ is then determined self consistently by computing the $\bar{s}$ for which $\langle \delta s^i(t)\rangle_T$ is zero, and setting $\langle s\rangle_T = \bar{s}$. The second moment $\langle \delta s^i(t)\delta s^j(t')\rangle_T$ will then be computed by using the expressions for $\langle s\rangle_T$ and Δ_T.

We can express the *effective path density* as

$$\begin{aligned}\mathcal{P}_{eff}[\tilde{s},\delta s] &= \exp(-S_{eff}[\tilde{s},\delta s]) \\ &= \exp\left(-S_F[\tilde{s},\delta s]\right)\left(\sum_{k=0}\frac{1}{k!}\left(-S_I[\tilde{s},\delta s]\right)^k\right),\end{aligned} \tag{7.61}$$

where

$$S_F[\tilde{s},\delta s] = \int_{t_0}^{\infty} \mathrm{d}t\ \mathrm{d}t'\tilde{s}(t)\Delta_T^{-1}(t,t')\delta s(t')$$

is the free part of the effective action, and

$$\begin{aligned}&S_I[\tilde{s},\delta s] \\ &= \int_{t_0}^{\infty}\mathrm{d}t'\tilde{s}_i(t')\mu^i(t') - \int_{t_0}^{\infty}\mathrm{d}t'\left(\sum_{n=2}^{\infty}\frac{1}{n!}\left(\frac{\tilde{s}_i(t')}{\tau}\right)^n\right)\left(\psi^i(t') + \kappa\, d\psi^i(t')\, W_j^i\,\delta s^j(t')\right)\end{aligned} \tag{7.62}$$

(compare with (7.56) for clarification). Here $\mu^i(t)$ is defined by

$$\mu^i(t) := \dot{\bar{s}}^i(t) + \frac{1}{\tau}\left(\bar{s}(t)^i - \psi^i(t)\right) - s_0^i\delta(t-t_0).$$

To compute $\langle \delta s^l(t)\rangle_T$, we must compute the path integral

$$\int \mathcal{D}\delta s\mathcal{D}\tilde{s}\ \ \delta s^l(t)\mathcal{P}_{eff}[\tilde{s},\delta s]. \tag{7.63}$$

This is accomplished by applying Wick's theorem 7.2.1 as in example 7.30. That is, we need to find all the terms in $S_I[\tilde{s},\delta s]$ which we can pair $\delta s^l(t)$ with in order to get a term with an equal number of $\tilde{s}$'s and δs's. For the mean, there are two such terms. First, we have

$$-\int_{t_0}^{\infty}\mathrm{d}t'\tilde{s}_i(t')\mu^i(t'), \tag{7.64}$$

and also we have, when $n = 2$,

$$\int_{t_0}^{\infty} \mathrm{d}t' \frac{1}{2} \left(\frac{\tilde{s}_i(t')}{\tau} \right)^2 \kappa \, d\psi^i(t') \, W_j^i \, \delta s^j(t'). \tag{7.65}$$

The second of these terms (7.65) will vanish however. This is because when pairing the $\tilde{s}$'s with s's as in Wick's theorem, we will always have a $\tilde{s}_i(t')$ paired with $\delta s^j(t')$. Since these terms are evaluated at the same time t', the resulting moment

$$\left\langle s^j(t') \tilde{s}_i(t') \right\rangle_F = \Delta_T(t', t')_i^j = 0.$$

This is due to the convention $H(0) = 0$ for the Heaviside function which results from the Ito interpretation of the PSM. Therefore, we find

$$\left\langle \delta s^l(t) \right\rangle_T = \int_{t_0}^{t} \mathrm{d}t' \Delta_T(t, t')_k^l \mu^k(t'). \tag{7.66}$$

For (7.66) to yield 0 for all t, we must have that $\mu^k(t) = 0$. Therefore the tree level mean must satisfy

$$\frac{\mathrm{d}}{\mathrm{d}t} \left\langle s^i \right\rangle_T = - \left\langle s^i \right\rangle_T + \phi \left(\kappa \sum_j W_j^i \left\langle s^j \right\rangle_T + b^i \right) + s_0^i \delta(t - t_0). \tag{7.67}$$

To compute $\left\langle \delta s^i(t) \delta s^j(t') \right\rangle_T$ we must compute the path integral

$$\int \mathcal{D}\delta s \mathcal{D}\tilde{s} \;\; \delta s^l(t) \delta s^j(t') \mathcal{P}_{eff}[\tilde{s}, \delta s]. \tag{7.68}$$

Again, we accomplish this computation by applying the expansion of the effective path density around (7.61) the free part of the action, and applying Wick's theorem by finding terms in the expansion with an equal number of s's and $\tilde{s}$'s. Notice that upon setting $\bar{s}^i(t) = \left\langle s^i(t) \right\rangle_T$, the μ term of the interacting action vanishes, and so it won't appear in the computation for the covariance. The only term in the interacting part of the action (7.62). that does couple to the covariance is the $\tilde{s}^2 \phi$ factor. It follows that to tree level

$$\left\langle \delta s^i(t) \delta s^j(t') \right\rangle_T = \int_{t_0}^{\infty} \mathrm{d}t_1 \Delta_T(t, t_1)_k^i \Delta_k^j(t', t_1) \phi(\kappa W_l^k \left\langle s^l(t_1) \right\rangle + b^k). \tag{7.69}$$

In the next chapter, I will apply a dimension reduction method to systems of the form (7.67). Section 8.5 will reconsider (7.69) in conjunction with the dimension reduction method in order get an analytic approximation the tree level covariances.

7.4 Appendix - functional derivatives

Suppose $F : \mathcal{F}(\mathbb{R}^N) \to \mathbb{R}$, where $\mathcal{F}(\mathbb{R}^N)$ is a space of functions over $\mathbb{R}^N$. I will refer to such an operator F as a *functional*. Given $f \in \mathcal{F}(\mathbb{R}^N)$, and $x \in \mathbb{R}^N$, the *functional derivative* [59] of F with respect to $f(x)$, written

$$\frac{\delta}{\delta f(x)} F(f), \tag{7.70}$$

is a type of derivative which satisfies the standard rules of linearity, and the Leibnitz rule. Additionally, when F is the identity operator we have

$$\frac{\delta}{\delta f(x)} f(y) = \delta(y - x),$$

where the $\delta(\cdot)$ is the Dirac delta function.

For example, say $f : \mathbb{R}^N \to \mathbb{R}$, and

$$F(f) := \int_{\mathbb{R}^N} \mathrm{d}y f(y)^2.$$

Then

$$\frac{\delta}{\delta f(x)} F(f) = \int_{\mathbb{R}^N} \mathrm{d}y \; 2f(y)\delta(y - x) = 2f(x).$$

Chapter 8

Low rank approximations of network activity

This chapter contains the main results of the thesis, and is the culmination of the material from the previous chapters. The primary focus is on a dimension reduction method for dynamics on large networks. The basic idea is to apply low rank approximations to all or parts of an adjacency matrix in order to infer aspects of dynamics on the network from a lower dimensional system. Employing the singular value decomposition (SVD) (see chapter 5) we perform a change of variables projecting the original dynamical variables onto a smaller set of variables which capture the essential aspects of the dynamics.

Suppose we have a network with adjacency matrix W, and there is some dynamical system defined on the network. That is, each node i has an associated dynamical variable x^i, and these variables interact according to the edges of W. Also, say we want to define a scalar quantity, and find an equation governing its evolution, which summarizes the activity across the population. Perhaps, the most obvious such variable is the mean. Given data, the mean is easy to compute, and is the least biased choice if one has no knowledge of the network. On the other hand, deriving an equation which governs the evolution of the mean entails the introduction of auxiliary variables, and an expansion in terms of the hierarchy of moments [4]. Often to get a closed set of equations one must truncate this hierarchy at some order. In some cases, for example if the network is completely connected, the mean

is a good representation of the dynamics. But if the network has heterogeneity then closing the equations for the mean may present difficulties.

The basic premise of the work presented in this chapter is that if one wants to represent the activity on a network with a small number of variables, it may be advantageous to consider the dynamics from the *network perspective*. By considering the topology of the network one may construct variables which characterize the flow of activity naturally. Of course, for very rich high dimensional dynamics one may not be able to effectively summarize activity across the population with several variables. But if the structure of the network has a high degree of randomness, we will see that it may be possible to find a small number of variables which essentially capture the dynamics. In fact, we will see that these variables may essentially recover the full activity of the population using the network, and this small set of dynamical variables.

The equations I will work with have the same form as the tree level mean, and covariance associated with the Poisson spiking model (PSM) derived in section 7.3. By applying the dimension reduction techniques in this chapter to those equations, we derive analytic expressions for population statistics which are parameterized by the in and out degree sequences. The equation for the covariances of activity over an ensemble (8.36) is similar, at least in spirit, to equations derived by Hu et al [29]. One of the benefits of our approach, though, is that in addition to deriving statistics for the activity across the network, we can directly recover the dynamics of the original system using the degree sequences and the derived population wide statistics. Nevertheless, a full comparison between the results and methods there and our approach would likely benefit both perspectives. That is left for future work. Also, it is likely that the reduction method I'll present can be applied to more general systems, but realizing the scope of its application is left open for future research.

Before diving into the equations, I will review some of the conventions regarding notation. In the first part of this chapter, I will present reduction techniques applied to differential equations over a network which have the same form as the tree level mean (7.67). In section 8.1, I will show how one can use a truncation of the SVD of the adjacency matrix to project the dynamics down to a lower dimensional system. We call this method the *low rank reduction*. I'll then present a special case of

the low rank reduction in section 8.2, called the *drive reduction*, which exploits the similarity between the principle singular vectors for a random network, and the in and out degree sequences, as discussed in chapter 5. In sections 8.3, and 8.4, I will generalize the drive reduction method to networks consisting of multiple subpopulations. Finally, in section 8.5 I will apply the degree approximation to the tree level covariance of the Poisson spiking model (7.69) to exhibit an analytic approximation for the covariance of the activity on a random network.

Notation

A quick reminder regarding some of the notational conventions for vectors: the components of row vectors are indexed with subscripts and the components of column vectors are indexed with superscripts. Given a network, I will treat the in degree sequence as a column vector, and the out degree sequence as a row vector. Hence the in and out degrees of node i are written d^i_{in}, and d^{out}_i.

Also, throughout this chapter I will continue to employ the *Einstein summation convention*. If an index appears as a superscript and a subscript in the same expression, then it should be understood that the index is summed over the values it can assume, unless explicitly stated otherwise. Thus, for example, we have

$$u_i v^i = \sum_i u_i v^i = v^i u_i.$$

On the other hand, there is no implied sum in an expression like $u^i v^i$.

8.1 Low rank reduction

Suppose we are given an $N \times N$ adjacency matrix W and a dynamical system of the form

$$\dot{x}^i = \gamma x^i + \phi\left(\kappa W^i_j x^j\right). \tag{8.1}$$

(Note the sum over j in (8.1)).

Let $W = U\Sigma V$ be the SVD of W, as in chapter 5, *except here we will assume that the right singular vectors form the rows of V instead of the columns.* Recall that the matrix Σ is diagonal, with non-negative singular values $\sigma^1 \geq \ldots \geq \sigma^N \geq 0$.

The columns of U are called the left singular vectors of W, and the *rows* of V are the right singular vectors. Let us write U and V as

$$U = \begin{pmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_N \\ | & | & & | \end{pmatrix}, \qquad \text{and} \qquad V = \begin{pmatrix} — & v^1 & — \\ — & v^2 & — \\ & \vdots & \\ — & v^N & — \end{pmatrix}.$$

Note that the i^{th} column vector of U is u_i. Since it is a column vector, the indices for the components of u_i will appear as superscripts. Therefore, u_i^j refers to the j^{th} component of the i^{th} left singular vector. Likewise v_k^i is the k^{th} component of the i^{th} right singular vector.

With this notation we may write

$$W = u_k \sigma^k v^k.$$

The rank l_0 approximation of W is given by

$$W_{l_0} = \sum_{k=1}^{l_0} u_k \sigma^k v^k. \tag{8.2}$$

The change of variable we propose for (8.1) is

$$x \mapsto \Sigma V x =: \hat{x},$$

or in terms of coordinates

$$\hat{x}^i = \sigma^i v_j^i x^j.$$

Thus the i^{th} component of $\hat{x}$ is taken to be the scaled dot product between the i^{th} right singular vector and x. This change of variable yields the system (sum over k and j)

$$\dot{\hat{x}}^i = \gamma \hat{x}^i + \sigma^i v_k^i \phi\left(\kappa u_j^k \hat{x}^j\right). \tag{8.3}$$

Intuitively, this change of variables separates the various features of W in order of their prominence. As long as all the coupling takes the form Wx, the system corresponding to the $\hat{x}$ variables will be driven most strongly by $\hat{x}^1$, and the least

by $\hat{x}^N$, where N is the number of nodes in the population.

Suppose we deem that the approximation $W_{l_0} \approx W$ is good for some l_0 (I don't have a universal criterion for what "good" means here). In this case, one may elect to use only the first l_0 singular values/vectors. This results in the *low rank reduction* (LRR) of (8.1)

$$\dot{\hat{x}}^i \approx \gamma \hat{x}^i + \sum_{k=1}^{l_0} \sigma^i v_k^i \phi \left(\kappa \sum_{j=1}^{l_0} u_j^k \hat{x}^j \right), \tag{8.4}$$

where $1 \leq i \leq l_0$. Notice that this l_0 dimensional system is closed (self-contained). Thus we have obtained a reduced closed system representing the original differential equations. Moreover, the form of these equations is not significantly more complicated than the original equation. In particular, it was not necessary to take derivatives of ϕ in order to close the equations.

In terms of the full system (8.1) the LRR corresponds to replacing W with W_{l_0}:

$$\dot{x}^i \approx \gamma x^i + \phi \left(\kappa \sum_{j=1}^{N} (W_{l_0})_j^i x^j \right). \tag{8.5}$$

Applying the change of variable $x \mapsto \hat{x}$ to the x's inside of ϕ in (8.5) yields the *decoupled system sourced by the LRR*

$$\dot{x}^i \approx \gamma x^i + \phi \left(\kappa \sum_{k=1}^{l_0} u_k^i \hat{x}^k \right). \tag{8.6}$$

The utility of (8.6) is that the dynamics of the full system may be effectively recovered from a lower dimensional system. Note: the only assumption for the low rank reduction is that the dynamics can be effectively projected onto an l_0 dimensional space. In the examples below I will illustrate the LRR. In particular, example 8.4.1 illustrates the application of the LRR to a system with oscillatory dynamics.

8.2 Drive reduction

In this section, I will present the *drive reduction*, which is a special case of the LRR introduced in the previous section. The drive reduction results in a one dimensional

system ((8.9) below).

The assumptions for applying the drive reduction method are:

1. **Weak coupling:** we'll assume that the coupling constant κ scales like $1/N$, where N is the number of nodes.

2. **The in and out degree sequences dominate the structure of W:** all correlations in network structure beyond second order are negligible.

As I argued in chapter 5, if the network has negligible correlations in its structure beyond second order, the outer product of the degree vectors furnishes a good rank one approximation to W. Recall that we define the *degree approximation* to W by

$$P = \frac{d_{\text{in}} d^{\text{out}}}{|W|}, \tag{8.7}$$

and also that $d_{\text{in}} d^{\text{out}}$ is an outer product since d_{in} is a column vector, and d^{out} is a row vector. I will refer to the LRR method as the *drive reduction* (DR) method when the degree approximation to W is applied. I will refer to the scalar variable

$$\hat{x} = \frac{1}{N \langle d \rangle} d_i^{\text{out}} x^i,$$

as the *drive* of x. The *decoupled rate equations sourced by the drive* variables (compare with (8.5) above) take the form

$$\dot{x}^i = \gamma x^i + \phi(d_{\text{in}}^i \hat{x}). \tag{8.8}$$

For the DR, (8.3) becomes a one dimensional system for the drive, $\hat{x}$

$$\dot{\hat{x}} = \gamma \hat{x} + \frac{1}{N \langle d \rangle} d_i^{\text{out}} \phi(d_{\text{in}}^i \hat{x}). \tag{8.9}$$

This equation is remarkable for a couple of reasons. Firstly, it is a single equation characterizing the population activity, and it is a complete closed equation. This is in contrast to an equation for the mean of the population, which does not close so easily. An equation for mean activity will generally involve an infinite hierarchy of moments [4], which needs to be truncated. By applying the degree approximation

to W, we effectively apply the averaging to the network and achieve a simple closed equation.

Another benefit of (8.9) is that it decouples the dynamics of x^i via (8.8). In a sense, the DR and the decoupled system represent the full dynamics of the original system in terms of local topology of the network (the degree sequence) and a single dynamical drive variable.

Example 8.2.1. In this example, I'll present anecdotal evidence for the veracity of the drive reduction, applied to the rate equation (7.67) for the Poisson spiking model (PSM). Even though this example is concerned with a single fixed network, the random nature of the network makes this example fairly generic. The phenomenon illustrated is highly typical of dynamics one sees for random networks generated by the expected degree model, and the SONET model (introduced in chapter 3) with similar values of p, and N.

The full rate equation found in the previous chapter is

$$\tau \dot{s}^i = -s^i + \phi(\kappa W^i_j s^j + b). \tag{8.10}$$

We will take the nonlinearity to be the scaled logistic function

$$\phi(x) = r/(1 + \exp(-x)).$$

The drive variable $\hat{s}$ associated with (8.10) satisfies

$$\tau \dot{\hat{s}} = -\hat{s} + \sum_i \frac{d^{\text{out}}_i}{N \langle d \rangle} \phi(\kappa d^i_{\text{in}} \hat{s} + b), \tag{8.11}$$

and the decoupled equations sourced by the drive variable are

$$\tau \dot{s^i} = -s^i + \phi(\kappa d^i_{\text{in}} \hat{s} + b). \tag{8.12}$$

Figure 8.2 shows a comparison between the steady state values of the full rate equations (8.10), and the decoupled equation source by the drive (8.12). The adjacency matrix for the network is depicted in figure 8.1. For this example the network was generated by the SONET model reviewed in section 3.1.2. In figure 8.2, one can

see that the agreement between the steady state values of the full equations (blue), and the decoupled equations (red) sourced by the drive is very good. Notice that the node indices are sorted according increasing in degree, but the steady state values for both the full equations and the decoupled equations deviate from being monotonically increasing, and the two systems largely agree in this respect as well.

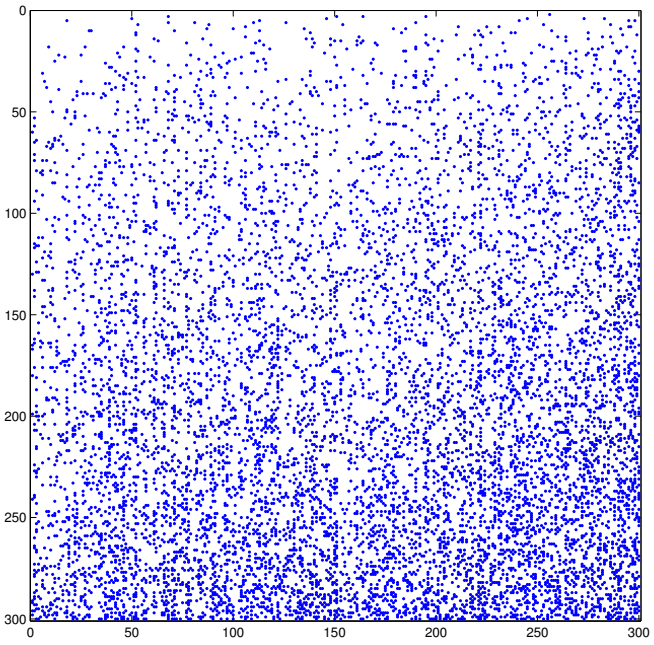


Figure 8.1: **Random adjacency matrix** This is a representation of a random adjacency matrix. A blue dot at the $(i, j)^{th}$ coordinate implies an edge from j to i. The network was sampled from the SONET model reviewed in section 3.1.2. The input parameters were $N = 300$, and $p = 0.1$, $\alpha_{\text{rcp}} = 0$, $\alpha_{\text{cnv}} = .3$, $\alpha_{\text{div}} = .2$, and $\alpha_{\text{chn}} = .1$.

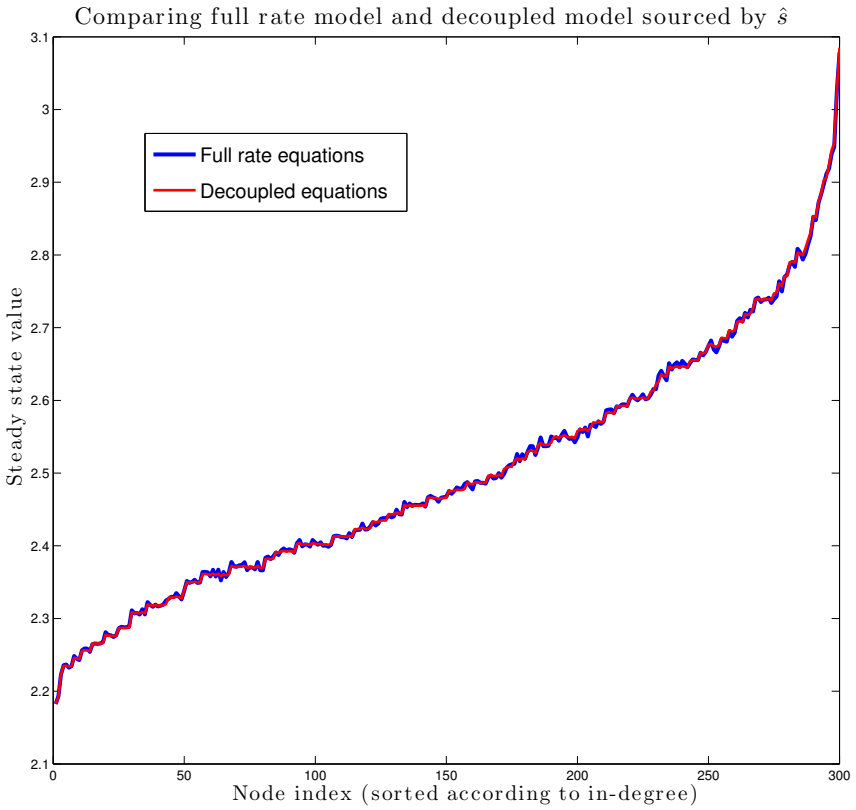


Figure 8.2: **Steady state firing rates** for a single excitatory population with adjacency matrix as in figure 8.1. The blue curve depicts the steady state values of the full rate equation, and the red curve consists of the steady state values of the decoupled rate equations sourced by the drive $\hat{x}$, as in (8.12). For this simulation $\phi(x) = r/(1 + \exp(-x))$, $r = 5$, $\kappa = 1/N = 1/300$, and the value of b was chosen so that the average value of the steady state across the population was $r/2 = 2.5$, the steepest part of the nonlinearity ϕ.

8.3 The drive reduction for two populations

Before delving into the general case of multiple populations, I will present the drive reduction for a system with two populations. I will refer to one of the populations as excitatory and the other as inhibitory, but they need not be of different types. I will refer to the network as an EI network simply to emphasize the two populations.

Covering the two population case will acquaint the reader with the basic idea of the method, and make the general case easier to grasp. I'll conclude the section with a pair of examples. The first example will show that the drive reduction can faithfully capture oscillations in the rate equation for the PSM. The second example will show the necessity of the weak coupling assumption (the first assumption in section 8.2) for the applicability of the drive reduction.

The second assumption for the applicability of the drive reduction in section 8.2 (that the degree sequences are the dominant structure of the network, i.e. all correlations in the structure beyond second order are negligible) must be modified for the two (and multiple) population case. The modified assumption is that for each pair of populations A and B (they may be the same), the submatrix W_A^B of the adjacency matrix corresponding to the collection of edges from a population A to a population B, is dominated by its degree sequences. As such, the degree approximation for W_A^B is a good rank-1 approximation in the sense of chapter 5.

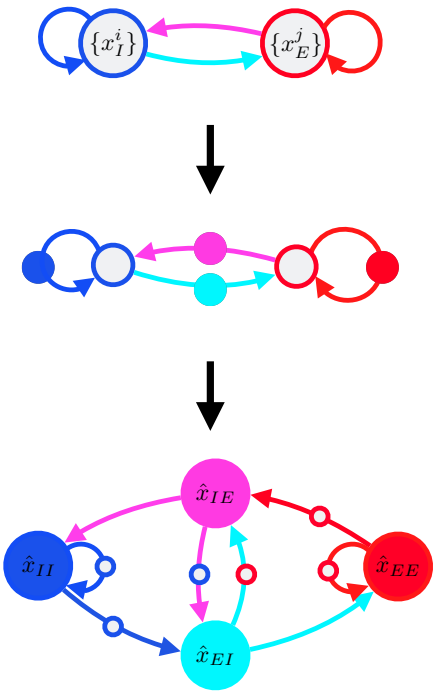


Figure 8.3: **Constructing the drive variables for an EI network** In the top diagram we have a population I with variables x_I^i $(i = 1, \ldots, N_I)$, and a population E with variables x_E^j $(j = 1, \ldots, N_E)$. Each edge of the top diagram in the figure represents a collection of edges from one population to another. For each of these collections of edges we will have an associated drive variable. The second diagram of the figure illustrates the creation of a drive variable for each of the macro scale edges, and the bottom diagram of the figure shows the directions of influence that the four drive variables $\hat{x}_{II}$, $\hat{x}_{IE}$, $\hat{x}_{EI}$, and $\hat{x}_{EE}$ have on one another. In going from the second to third diagrams in the figure, we find that each of the populations yields edges between the drive variables. These *induced* connections are indicated in the bottom diagram by small copies of the original populations (gray filled dots). Thus, for example, the I population in the second diagram yields the edge from the drive $\hat{x}_{IE}$ to the drive $\hat{x}_{EI}$. Note that the indices of the drive variables IE, and EI should be read right to left. So IE represents the collection of edges from E to I.

Figure 8.3 illustrates the drive reduction for a two population system. First, let us express the full system in terms of the variables x_E^j, and x_I^i, in the top diagram of figure 8.3. The full system corresponding to (8.1) can be expressed in terms of

the population variables as

$$\begin{aligned}\dot{x}_E^j &= \gamma x_E^j + \phi\left(\kappa_{EE} W_{E,k}^{E,j} x_E^k + \kappa_{EI} W_{I,l}^{E,j} x_I^l + b_E\right) \\ \dot{x}_I^i &= \gamma x_I^i + \phi\left(\kappa_{IE} W_{E,k}^{I,i} x_E^k + \kappa_{II} W_{I,l}^{I,i} x_I^l + b_I\right),\end{aligned} \tag{8.13}$$

where $W_{A,i}^{B,j} \neq 0$ if there is an edge from node i of population A to node j of population B, and sums are only over node indices (e.g. k, and l).

Let $|d_{\text{in}}^j(B,A)|$ be the number of edges from population A which point to node j of population B, and set $d_{\text{in}}^j(B,A) \geq 0$ if A is excitatory, and $d_{\text{in}}^j(B,A) \leq 0$ if B is inhibitory. Similarly, let $d_i^{\text{out}}(B,A)$ be the number of outgoing edges from node i in population A which are directed to nodes in B. Note: in this formulation $d^{\text{out}}(B,A)$ is always nonnegative. The sign of W_A^B is carried by the $d_{\text{in}}(B,A)$ vector. With this notation the degree approximation (see (8.7)) to W_A^B is

$$W_A^B = \frac{1}{|W_A^B|} d_{\text{in}}(B,A) d^{\text{out}}(B,A), \tag{8.14}$$

where $|W_A^B|$ is the total number of edges from A to B.

We may now define the *drive variables* associated with the two population system

$$\hat{x}_{BA} = \frac{1}{|W_A^B|} d_i^{\text{out}}(B,A) x_A^i. \tag{8.15}$$

The drive variable $\hat{x}_{BA}$ is the total output of the population A onto population B.

Applying the degree approximation to (8.13), and using the definition of the drive variables yields the *decoupled system sourced by the drive variables*

$$\begin{aligned}\dot{x}_E^j &= \gamma x_E^j + \phi\left(\kappa_{EE} d_{\text{in}}^j(E,E)\hat{x}_{EE} + \kappa_{EI} d_{\text{in}}^j(E,I)\hat{x}_{EI} + b_E\right) \\ \dot{x}_I^i &= \gamma x_I^i + \phi\left(\kappa_{IE} d_{\text{in}}^i(I,E)\hat{x}_{IE} + \kappa_{II} d_{\text{in}}^i(I,I)\hat{x}_{II} + b_I\right).\end{aligned} \tag{8.16}$$

Finally, the drive variables evolve according to the system

$$\begin{aligned}
\dot{\hat{x}}_{EE} &= \gamma \hat{x}_{EE} + \frac{1}{|W_E^E|} d_j^{\text{out}}(E,E)\phi\left(\kappa_{EE} d_{\text{in}}^j(E,E)\hat{x}_{EE} + \kappa_{EI} d_{\text{in}}^j(E,I)\hat{x}_{EI}\right) \\
\dot{\hat{x}}_{IE} &= \gamma \hat{x}_{IE} + \frac{1}{|W_E^I|} d_j^{\text{out}}(I,E)\phi\left(\kappa_{EE} d_{\text{in}}^j(E,E)\hat{x}_{EE} + \kappa_{EI} d_{\text{in}}^j(E,I)\hat{x}_{EI}\right) \\
\dot{\hat{x}}_{EI} &= \gamma \hat{x}_{EI} + \frac{1}{|W_I^E|} d_i^{\text{out}}(E,I)\phi\left(\kappa_{IE} d_{\text{in}}^i(I,E)\hat{x}_{IE} + \kappa_{II} d_{\text{in}}^i(I,I)\hat{x}_{II}\right) \\
\dot{\hat{x}}_{II} &= \gamma \hat{x}_{II} + \frac{1}{|W_I^I|} d_i^{\text{out}}(I,I)\phi\left(\kappa_{IE} d_{\text{in}}^i(I,E)\hat{x}_{IE} + \kappa_{II} d_{\text{in}}^i(I,I)\hat{x}_{II}\right).
\end{aligned} \tag{8.17}$$

Note that (8.17) is a four dimensional system. In a mean field approach one would typically have two equations for an EI network. It remains a matter for future investigation to find a network and parameters where the four dimensional equations exhibit behavior not achievable with two dimensions. It would also be a nice result to show that for some networks one needs a four dimensional system to recover the full dynamics as in (8.17).

Example 8.3.1. In this example, I'll present further anecdotal evidence for the veracity of the drive reduction, applied to the rate equation (8.18) for the PSM, considered in example 8.2.1. We will see that the drive reduction can both capture the oscillatory dynamics, and reproduce the population dynamics via the decoupled equations sourced by the drive variables (8.16). As in example 8.2.1 the results shown in figure 8.4 are representative of what one sees generically with the drive reduction for two populations using a weakly coupled rate equation for the PSM. I will also show a single trial of the PSM (introduced in example 6.2.1), to give some idea of how well the rate equation reflects the activity of the spiking network.

The rate equations for two populations read

$$\begin{aligned}
\tau_E \dot{s}_E^j &= -s_E^j + \phi(\kappa_{EE} W_{E,k}^{E,j} s_E^k + \kappa_{EI} W_{I,l}^{E,j} s_I^l + b_E) \\
\tau_I \dot{s}_I^i &= -s_I^i + \phi(\kappa_{IE} W_{E,k}^{I,i} s_E^k + \kappa_{II} W_{I,l}^{I,i} s_I^l + b_I),
\end{aligned} \tag{8.18}$$

where, again, the nonlinearity is the scaled logistic function

$$\phi(x) = r/(1+\exp(-x)).$$

We now have four drive variables $\hat{s}_{EE}, \hat{s}_{EI}, \hat{s}_{IE}$, and $\hat{s}_{II}$, which satisfy the four dimensional system

$$\tau \dot{\hat{s}} = -\hat{s} + \sum_i \frac{d_i^{\text{out}}}{N \langle d \rangle} \phi(\kappa d_{\text{in}}^i \hat{s} + b), \tag{8.19}$$

and the decoupled equations sourced by the drive variable are

$$\begin{aligned}
\tau_E \dot{\hat{s}}_{EE} &= -\hat{s}_{EE} + \frac{1}{|W_E^E|} d_j^{\text{out}}(E,E)\phi\left(\kappa_{EE} d_{\text{in}}^j(E,E)\hat{s}_{EE} + \kappa_{EI} d_{\text{in}}^j(E,I)\hat{s}_{EI} + b_E\right) \\
\tau_E \dot{\hat{s}}_{IE} &= -\hat{s}_{IE} + \frac{1}{|W_E^I|} d_j^{\text{out}}(I,E)\phi\left(\kappa_{EE} d_{\text{in}}^j(E,E)\hat{s}_{EE} + \kappa_{EI} d_{\text{in}}^j(E,I)\hat{s}_{EI} + b_E\right) \\
\tau_I \dot{\hat{s}}_{EI} &= -\hat{s}_{EI} + \frac{1}{|W_I^E|} d_i^{\text{out}}(E,I)\phi\left(\kappa_{IE} d_{\text{in}}^i(I,E)\hat{s}_{IE} + \kappa_{II} d_{\text{in}}^i(I,I)\hat{s}_{II} + b_I\right) \\
\tau_I \dot{\hat{s}}_{II} &= -\hat{s}_{II} + \frac{1}{|W_I^I|} d_i^{\text{out}}(I,I)\phi\left(\kappa_{IE} d_{\text{in}}^i(I,E)\hat{s}_{IE} + \kappa_{II} d_{\text{in}}^i(I,I)\hat{s}_{II} + b_I\right).
\end{aligned} \tag{8.20}$$

Figure 8.4 shows results of the above equations. In particular, we find that the full equations sourced by the drive variables faithfully replicates the dynamics on the full network.

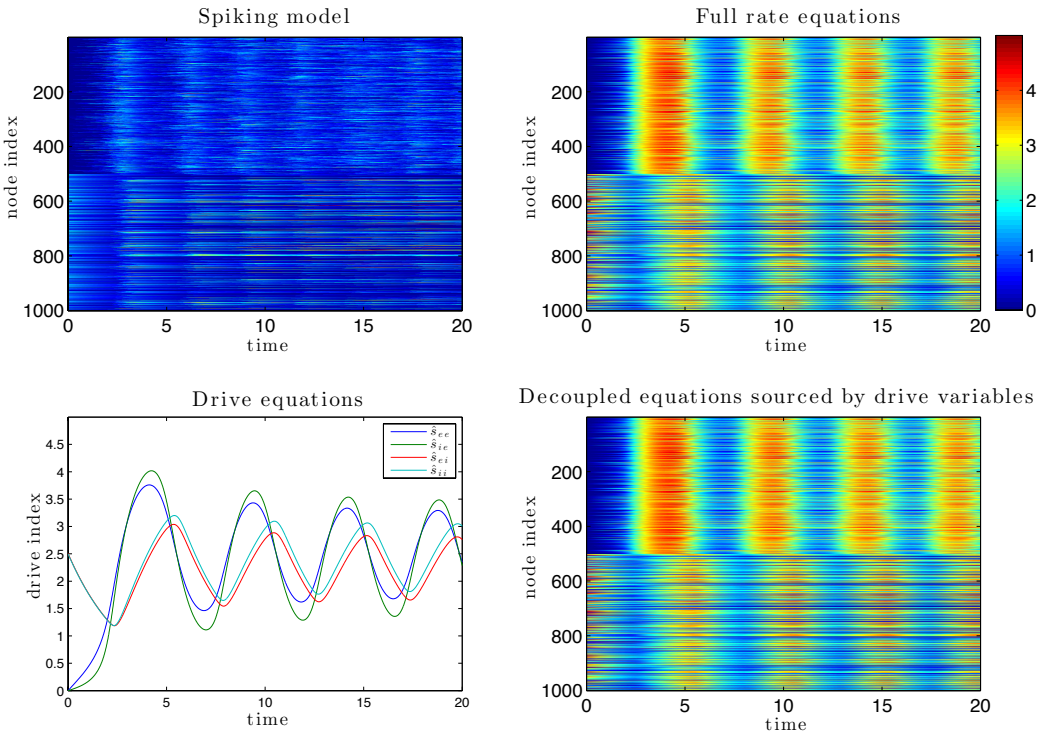


Figure 8.4: **Capturing the oscillations of an EI network with the drive reduction** For the above plots, the network consists of 500 excitatory nodes (indexed 1, through 500), and 500 inhibitory nodes (indexed 501 to 1000). Each of the submatrices W_B^A, where A and B are E's and I's was generated using the SONET model, with varying densities and second order statistics, but with all chains negative. The time constants were $\tau_E = 1$, and $\tau_I = 3$. Each of the coupling constants κ was between 0.01, and 0.02. The plot titles explain what each plot is. The color scale on both images to the ***right*** are the same. Notice that the images to the right look very similar.

In the following example, we will examine a case in which the decoupled system sourced by the drive variables (8.16) fails to capture even qualitative behavior of the full system (8.13). Thankfully, this failure relies on violating the weak coupling assumption.

Example 8.3.2. In this example, I will illustrate a way in which the drive reduction method may fail if we drop the weak coupling assumption. In their famous paper, Sompolinsky, Crisanti, and Sommers [46] show that large random networks may have

chaotic dynamics. In particular, we can exhibit chaos in a network with excitatory and inhibitory subpopulations and random connectivity. In this setting, we will have four drive variables, as above, and while it may be possible to have chaos in three dimensional ODE (e.g. the Lorentz attractor) it does not seem possible with the system in this example. As such, the decoupled equations sourced by the drive variables must fail to capture the dynamics of the full system in this setting.

The system considered in [46] is

$$\dot{x}^i = -x^i + J^i_j \tanh(gx^j), \tag{8.21}$$

where the components of J^i_j are independent and distributed according to a normal distribution with mean 0, and variance J^2/N. What they find is that for $gJ > 1$ the system is chaotic, in the large N limit. Note that if J is relatively small so that coupling is weak, the gain g must be large to compensate.

For this example we consider a modified system. In [53], they indicate that chaos can be induced in populations satisfying Dale's principle (nodes are either excitatory or inhibitory). Also, I have found through numerical experiments that one can find chaotic dynamics in a system where excitatory and inhibitory populations are connected with pairs of nodes connected with probability p, and that is the connectivity we will presently consider. For this example we will have an excitatory population and an inhibitory population, each with the same number of nodes N. And we will fix a probability $p \in (0, 1)$. For each of the pairs $(A, B) \in \{(E, E), (E, I), (I, E), (I, I)\}$ let $\pm W^A_B$ be an (signed) adjacency matrix for an Erdös-Réyni network with edge density p. If $B = E$ then we take the sign to be positive; If $B = I$ we take it to be negative. The form of the dynamics I'll use are

$$\begin{aligned} \dot{x}^j &= -x^j + \tanh\left(gW^{E,j}_{E,l}x^l + gW^{E,j}_{I,k}y^k\right) \\ \dot{y}^i &= -y^i + \tanh\left(gW^{I,i}_{E,l}x^l + gW^{I,i}_{I,k}y^k\right). \end{aligned} \tag{8.22}$$

These equations are exactly the same as those derived above in this section, taking $\gamma = -1$, and $\phi(x) = \tanh(x)$.

In figure 8.5, we see failure of the drive reduction to capture the chaos of the full system. As noted above, to get chaos in the full system requires the coupling

parameter g to be fairly large. When the coupling is strong the specifics of the connectivity are significant because perturbations in one variable may cause noticeable differences in the trajectory of its neighbors. In this case the frozen noise in the network plays an important role in the dynamics. If one were to apply the low rank reduction (using the SVD) it may require a greater number of singular vectors to capture the noise of the system. On the other hand, in the weakly coupled case the system may essentially decouple via the drive reduction (8.16).

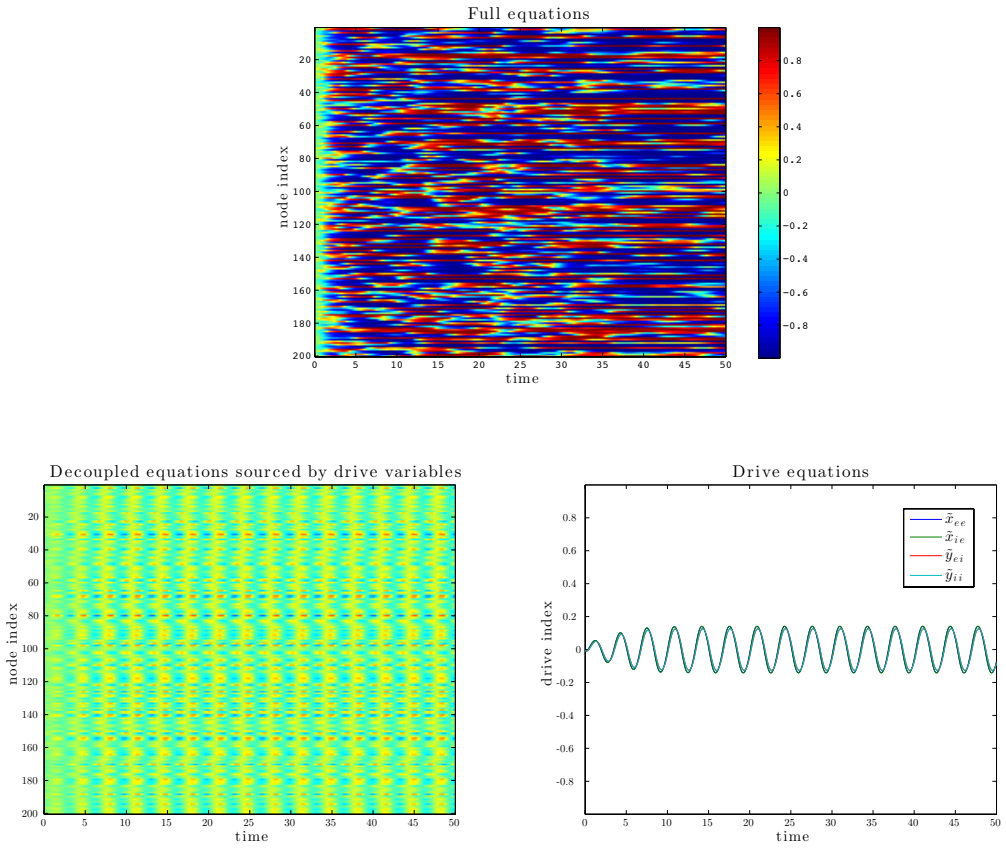


Figure 8.5: **Chaos** For the above plots, the network consists of 100 excitatory and 100 inhibitory nodes. Each edge of the network was independently generated with probability $p = 0.15$. The parameter g was chosen to make the product $gJ = 4$, (above the chaos boundary of $gJ = 1$). In the ***top*** image is the result of numerically integrating (8.22) with the full network. The ***lower left*** is the result of the integrating the decoupled equations sourced by the drive variables, and the ***lower right*** plot shows the drive variables, which are periodic and synchronous. The color scales for the top, and lower left plots are the same.

8.4 The drive reduction for multiple populations

The brain can be viewed as being modular and consisting of many subpopulations which are interconnected. To understand the interplay between large scale dynamics and cellular level dynamics it may be useful to have a technique for reducing

the dimension of a system while preserving the population level structure. In this section, I will generalize the results of the previous section to an arbitrary number of populations.

Suppose we have N_{pop} populations connected according to some macro structure, as in the constructions in chapter 4. For example, we could have populations arranged around a ring as in example 4.5.2. I will use the letters $a, b, c, a_1, \ldots$ to index subpopulations, and $i, j, k, i_1 \ldots$ for node indices. Specifically, say $\hat{W}$ is an $N_{pop} \times N_{pop}$ adjacency matrix for the macro scale structure of a network. Then $\hat{W}_b^a = 1$ if there are any edges from population b to population a. The adjacency matrix connecting population b to a will be denoted W_b^a. If there is an edge from the j^{th} node of the b^{th} population to the i^{th} node of the a^{th} population then we will have that $W_{b,j}^{a,i} = 1$.The full system (8.1), can be written

$$\dot{x}^{a,i} = \gamma x^{a,i} + \phi \left(\sum_{b,j} \kappa_b^a W_{b,j}^{a,i} x^{b,j} \right). \tag{8.23}$$

As in the two population case in section 8.3, we will assume that the adjacency matrices connecting subpopulations are each well approximated by the outer product of the degree sequences. That is, the following approximation is applicable

$$W_b^a \approx \frac{(d_{\text{in}})_b^a (d^{\text{out}})_b^a}{|W_b^a|},$$

where $|(d_{\text{in}})_b^{a,i}|$ is the number of edges node i in population a receives from population b, and $(d^{\text{out}})_{b,j}^a$ is the number of outgoing edges from node j in population b which go to population a. (Recall, since d_{in} is a column vector, and d^{out} is row vector, $d_{\text{in}} d^{\text{out}}$ is an outer product.) As with two populations, if W_b^a has negative entries we will follow the convention that $(d^{\text{out}})_b^a$ is positive, and $(d_{\text{in}})_b^a$ is negative. As a rule, d_{in} will carry the sign of W.

We will have a drive variable associated with each edge of $\hat{W}$. Let $\hat{x}_b^a$ be the drive of x across the edges from population b to population a. The decoupled equations sourced by $\hat{x}$ are

$$\dot{x}^{a,i} = \gamma x^{a,i} + \phi \left(\sum_{b} \kappa_b^a (d_{\text{in}})_b^{a,i} \hat{x}_b^a \right). \tag{8.24}$$

The drive equations are

$$\dot{\hat{x}}_b^a = \gamma \hat{x}_b^a + \sum_j \frac{(d^{\text{out}})_{b,j}^a}{|W_b^a|} \phi \left(\sum_b \kappa_b^a (d_{\text{in}})_b^{a,j} \hat{x}_b^a \right). \tag{8.25}$$

The dimension of the reduced system (8.25) is equal to the number of edges of $\hat{W}$.

Note: Given dynamics on an (possibly signed) adjacency matrix W for a network with subpopulations connected according to some macroscopic network one could apply the low rank reduction in two different ways:

1. One could take the SVD of the entire adjacency matrix, and proceed as in section 8.1, or

2. One could take the SVD of each submatrix of the adjacency matrix corresponding to a macro scale edge. That is, if there are connections from population A to population B, then one could take the SVD of the submatrix W_A^B and apply the low rank reduction for that submatrix, and all other submatrices corresponding to edges of the macroscopic structure. This is a generalization of the reductions in (8.24), and (8.25) using the low rank reduction.

As we will see in the next example, the first scenario can give qualitatively good results. On the other hand, the second case might be preferable for a few reasons. If the size of the network makes computing the SVD of the entire adjacency matrix prohibitive, it may be more computationally tractable to apply the SVD only to connections between subpopulations. Moreover, preserving the macroscopic structure in its entirety could lead to more accurate results. A low rank approximation (via the SVD, see (8.2)) to an (binary, or signed binary) adjacency matrix will generally have nonzero entries. In other words the low rank approximation does not perfectly preserve the information that two nodes or two subpopulations are not connected. In the second scenario above, that information is preserved at the macro scale. A rigorous comparison between the two is a matter for future investigation.

Example 8.4.1. I will now illustrate both the LRR method, and DR method for a network with multiple subpopulations. We will see that both methods do a good job at recovering the oscillatory dynamics of the full rate equations. For this example, an EI network was generated, and parameters of the rate equation were tuned to

produce a traveling wave solution to the rate equations, shown in the upper right panel of figure 8.8. The macro scale network is depicted in figure 8.6. The caption of that figure describes how the full adjacency matrix shown in figure 8.7 was generated. Example 4.5.3 gives the details of the procedure. Also, I have included a single trial of the original PSM in the upper left panel of figure 8.8 for comparison.

The point of this example, beyond illustrating the reduction methods of this chapter, is to show how a traveling wave solution of the rate equations on a network with random connectivity, may be seen as arising from the macro scale symmetry of the network. In particular, we will see that the DR variables exhibit the symmetry of the macro scale structure.

It is well known that a symmetry of a network yields a symmetry of any ODE coupled via the network [26], but adjacency matrix for this example (figure 8.7) has no nontrivial symmetries (i.e. relabeling the nodes will change the network). Moreover, the full rate equations shown in the upper right panel of figure 8.8 have no obvious symmetries (i.e. permuting the indices would likely alter the image). Nevertheless, the traveling wave solution in figure 8.8, can be seen to be consistent with the rotational symmetry of the macro scale structure of the ring (figure 8.6) via the drive reduction. The network governing the evolution of the drive reduction variables (middle left panel of figure 8.8) is nearly symmetric as long as the connectivities between subpopulations are identically, independently distributed. As such, the asymmetric aspects of the full rate equations can be attributed to the micro scale randomness, but the traveling wave solution can be attributed to the macro scale symmetry of the network.

The bottom left panel of figure 8.8 tells a different story. These are the dynamics corresponding to the LRR. It appears that the most significant variable of the LRR (boldest red curve) acts as a basic offset for the activity of the network. The other curves, appear to be oscillations at different phases. A deeper analysis of these dynamics would require looking at the specific singular vectors, and how they encode the structure. That is beyond our current scope, and will have to wait for future research.

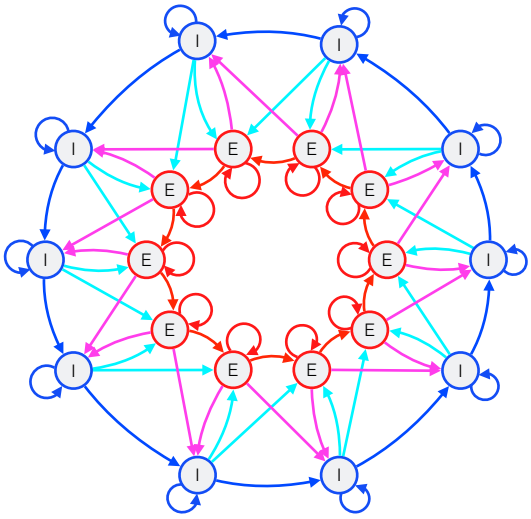


Figure 8.6: **Macro scale structure for E/I ring** This diagram shows the macro structure for the network in this example. The colors of the edges coincide with the colors of the full adjacency matrix in figure 8.7. Each edge of the macro scale network has an associated drive variable. The specific connectivity between each of the populations is depicted in the adjacency matrix in figure8.7. Notice that there are 8 edge types. Let $(a, b) \in \{(E, E), (E, I), (I, E), (I, I)\}$. Then the 8 types of edges occur as a may connect to the b situated along the same radial axis, or the b in the next pair around the circle. Each of the 8 types of edges has an associated probability p which was used to generate the edges within each block of figure 8.6.

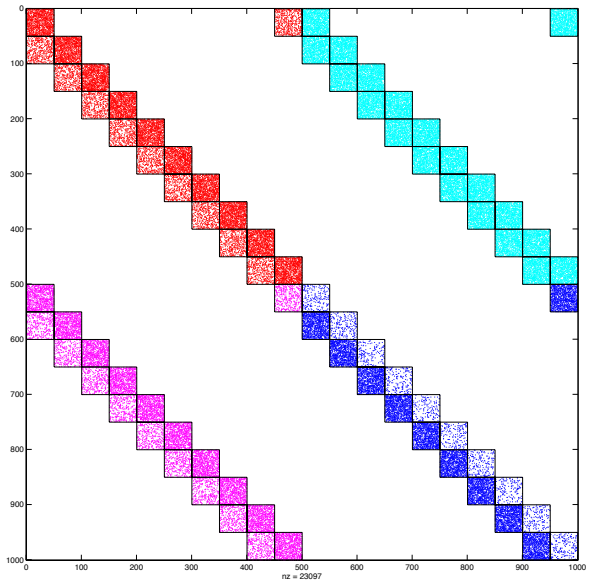


Figure 8.7: **Random E/I ring** This figure illustrates the adjacency matrix for this example. The dots are color coded as follows: Red - excitatory to excitatory, Magenta - excitatory to inhibitory, Cyan - inhibitory to excitatory, Blue - inhibitory to inhibitory. Each edge in figure 8.6 corresponds to a 50×50 square above. Moreover, each square corresponds to a single drive variable. The connectivities were randomly generated as in example 4.5.3 with varying probabilities for the 8 different types of edges appearing in figure 8.6.

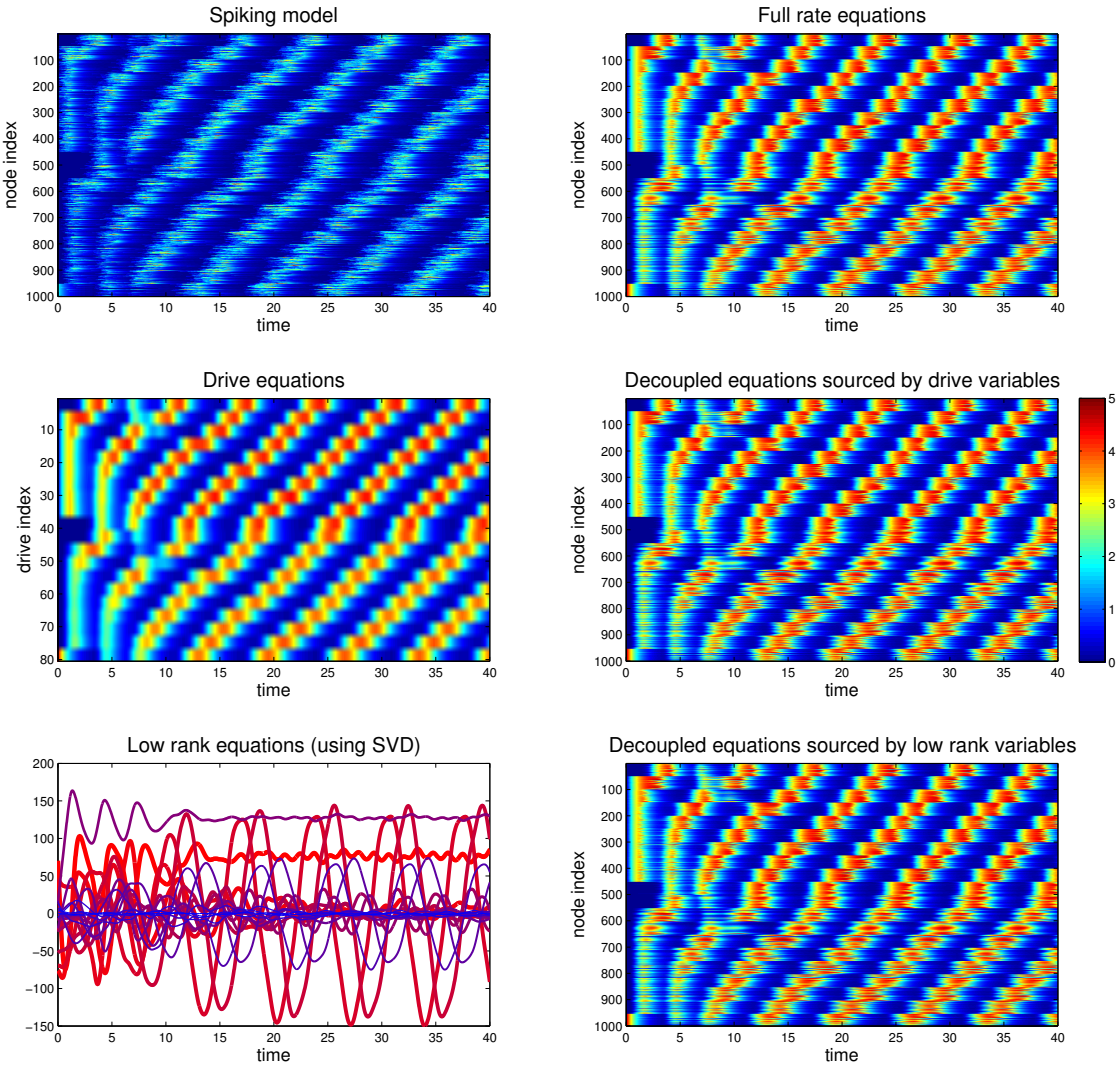


Figure 8.8: **Comparing multi population dynamics** In the ***upper left*** panel is a single trial of the PSM. ***Upper right***, we have the full rate equations (tree level mean) of the PSM simulated for the network in figure 8.7. The ***middle left*** panel shows the results of the drive reduction equations. Evidently, the drive reduction could be reduced further to 20 variables owing to the redundancy in the trajectories.The ***middle right*** panel shows the decoupled equations sourced by the drive reduction. Similarly, the ***lower left*** panel shows the trajectories of the low rank reduction. In this case the rank was taken to be 20. The boldest red curve corresponds to the highest rank variable, while the thinnest blue curve corresponds to the 20^{th} rank reduction variable. The ***lower right*** panel shows the decoupled system sourced by the low rank reduction. The color scale for all of the panels on the right as well as the middle left are as depicted in the color bar to the right.

8.5 Covariances of the PSM

In this section I will apply the degree approximation to W used in the DR to simplify the equations for the tree level covariances computed in section 7.3. Following the derivation, I will provide an example showing how the results compare with covariances computed directly from the PSM.

Recall, that the PSM satisfies

$$\tau \dot{s^i} = -s^i + \sum_k \delta(t - t_k^i),$$

where the spike times $\{t_k^i\}$ are generated according an inhomogeneous Poisson process with rate $\phi(\kappa W_j^i s^j + b^i)$. Define g

$$g^i = W_j^i s^j \approx \frac{d_{\text{in}}^i d_j^{\text{out}}}{|W|} s^j = d_{\text{in}}^i \hat{s}.$$

The equation for the tree level mean, or *rate equation*, was computed in section 7.3 (equation (7.67))

$$\tau \dot{\bar{s}}^i = -\bar{s}^i + \phi(\kappa W_j^i \bar{s}^j + b^i). \tag{8.26}$$

The tree level propagator (equation (7.60)) was found to satisfy

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta_j^i(t,t') = -\frac{1}{\tau}\Delta_j^i(t,t') + \frac{\kappa}{\tau}\phi'(\kappa \bar{g}^i(t) + b^i) W_k^i \Delta_j^k(t,t') + \delta_j^i \delta(t-t'). \tag{8.27}$$

Suppose the system is *strict-sense stationary*, so that its statistics are invariant with respect to shifts in time [39]. In this case, the mean field equation is in equilibrium so $\bar{g}(t)$ is constant. Then (8.27) has solution

$$\Delta_j^i(t,t') = \exp\left(-\Gamma(t-t')\right) H(t-t'), \tag{8.28}$$

where $H(t-t')$ is the Heaviside function (where we take $H(0) = 0$), and with

$$\Gamma_j^i = \frac{1}{\tau}\left(\delta_j^i - \kappa\phi'(\kappa \bar{g}^i + b^i) W_j^i\right). \tag{8.29}$$

The tree level covariance (from equation (7.69)) of s can be written as

$$\begin{aligned} C^{ij}(t+t',t') &:= \left\langle \delta s^i(t'+t)\delta s^j(t') \right\rangle_c \\ &= \int_{t_0}^{\min(t',t'+t)} \Delta^i_k(t'+t,t_1)\Delta^j_k(t',t_1)\phi(\kappa\bar{g}^k(t_1)+b^k)\mathrm{d}t_1. \end{aligned} \tag{8.30}$$

Since I will be chiefly concerned with the steady state statistics let $C^{ij}(t) := C^{ij}(t+t',t')$. Combining equations (8.27), and (8.30) we have

$$\frac{\mathrm{d}}{\mathrm{d}t}C^{ij}(t) = -\Gamma^i_k C^{kj} + \Delta^j_i(t',t+t')\phi(\kappa\bar{g}^i(t+t')+b^i). \tag{8.31}$$

Note, there is no sum in the last term of (8.31), and that $\Delta^j_i(t',t+t') \neq 0$ only if $t<0$. Let $D_\phi = \mathrm{Diag}(\phi(\kappa\bar{g}^i+b^i))$ be the diagonal matrix with the i^{th} diagonal entry equal to $\phi(\kappa\bar{g}^i+b^i)$. Using the fact that $\bar{g}$ is constant, and using the exponential form of $\Delta^i_j(t,t')$ we can write

$$\frac{\mathrm{d}}{\mathrm{d}t}C(t) = -\Gamma C(t) + H(-t)D_\phi \exp(t\Gamma^T), \tag{8.32}$$

or

$$\frac{\mathrm{d}}{\mathrm{d}t}\left[\exp(t\Gamma)C(t)\right] = H(-t)\exp(t\Gamma)D_\phi\exp(t\Gamma^T). \tag{8.33}$$

Let's now assume that W is essentially second order. That is, the approximation $W \approx \frac{d_{\mathrm{in}}d^{\mathrm{out}}}{N\langle d\rangle}$ is applicable. For convenience let $\hat{d_{\mathrm{in}}}^{\,i} = \phi'(\kappa\bar{g}^i+b^i)d^i_{\mathrm{in}}$, and $\hat{d^{\mathrm{out}}}_j = d^{\mathrm{out}}_j/N\langle d\rangle$. I want to use this assumption to simplify $\exp(t\Gamma)$, and solve (8.33). Letting $\beta_{chn} = \hat{d^{\mathrm{out}}}\cdot\hat{d_{\mathrm{in}}}$ we find (see section B below)

$$\Gamma^n \approx \frac{1}{\tau^n}\left(I - \frac{((1-\kappa\beta_{chn})^n-1)}{\beta_{chn}}\hat{d_{\mathrm{in}}}\hat{d^{\mathrm{out}}}\right).$$

Therefore,

$$\exp(t\Gamma) \approx \exp(t/\tau)\left(I + (1-\exp(-\kappa\beta_{chn}t/\tau))\frac{1}{\beta_{chn}}D_{\phi'}W\right). \tag{8.34}$$

Assuming steady state, and that $C(-\infty) = 0$ we have

$$\int_{-\infty}^{0} \frac{\mathrm{d}}{\mathrm{d}t} \left[\exp(t\Gamma)C(t)\right] \mathrm{d}t = \exp(t\Gamma)C(t)\Big|_{-\infty}^{0} = C(0). \tag{8.35}$$

Thus, integrating the right hand side of (8.33) from $-\infty$ to 0, will yield the equal time covariance. Plugging (8.34) into (8.33), and integrating yields

$$\begin{aligned} C(0) &= \int_{-\infty}^{0} H(-t) \exp(t\Gamma) D_\phi \exp(t\Gamma^T) \\ &= \frac{\tau}{2}\left(D_\phi + \frac{\kappa}{2 - \beta_{chn}\kappa}(D_{\phi'} W D_\phi + D_\phi W^T D_{\phi'}) \right. \\ &\qquad \left. + \frac{\kappa^2}{2 - 3\beta_{chn}\kappa + \beta_{chn}^2\kappa^2} D_{\phi'} W D_\phi W^T D_{\phi'} \right). \end{aligned} \tag{8.36}$$

We can use this as the initial condition for (8.32), which for $t > 0$ is simply the homogenous equation

$$\frac{\mathrm{d}}{\mathrm{d}t} C(t) = -\Gamma C(t). \tag{8.37}$$

Equation (8.37) is solved by

$$C(t) = \exp(-t\Gamma)C(0). \tag{8.38}$$

Example 8.5.1. This is a numerical check of the validity of the derived estimate for the steady state equal-time (co)variances (8.36), based on the degree approximation to W. For this example, I'll take a single excitatory population with nontrivial second order statistics. Figure 8.9 shows the result of an empirically computed variance compared with the result predicted by (8.36), and figure 8.10 shows a comparison of the analytically computed covariances with those computed from simulations. In both figures, there are trends which are captured, but there are also large errors. It is difficult to know how much of the error is due to the degree approximation used for W, how much is due to the tree level approximation for $C(0)$, and how much is due to trial to trial fluctuations. For example, in figure 8.10, the empirically estimated covariances (left) have a number of negative values. Even if we had computed the tree level covariance $C(0)$ with no approximations to W, all of the entries of $C(0)$ would be non-negative.

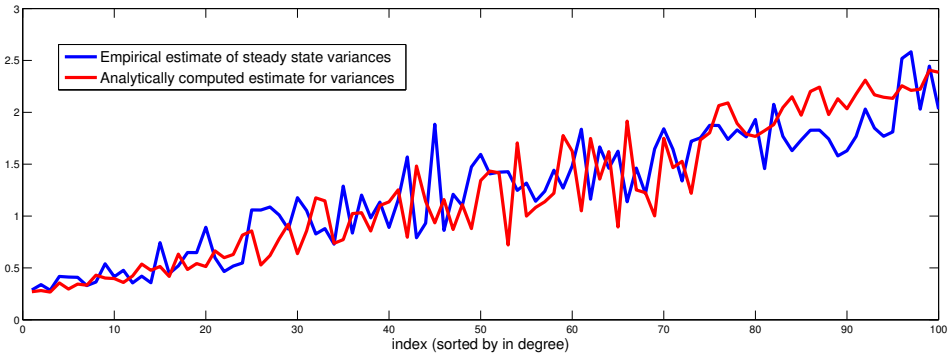


Figure 8.9: **Steady state variances for a purely excitatory population** The variances were estimated from 1000 trials. In this example W was constructed using a randomly generated expected degree model, as in example 3.3.1. The input parameters were $N = 100$, and $p = 0.1$. The produced network had observed second order statistics $\alpha_{\text{cnv}} = 0.12$, $\alpha_{\text{div}} = 0.1$, and $\alpha_{\text{chn}} = -0.02$. The coupling was $\kappa = 1/\sqrt{N} = 0.1$, and the rate function was $\phi(x) = r/(1 + \exp(-x))$, with $r = 5$. The network was driven to steady state with the average value of s^i across the population equal to 2.5, at the steepest point of the nonlinearity. The estimate captures the general trend, but is clearly not perfect. Possible sources of error are degree approximation of W, the tree level approximation to $C(0)$, or an insufficient number of trials.

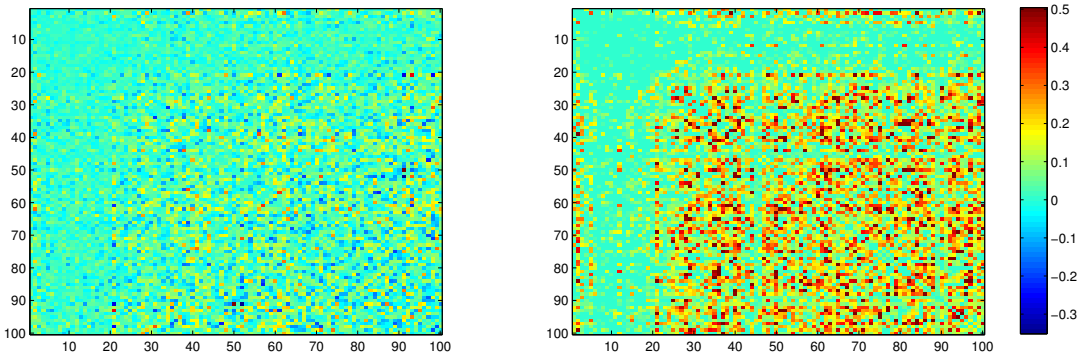


Figure 8.10: **Steady state covariances for a purely excitatory population** The ***left*** panel shows the equal time covariances empirically computed from the same simulations as those used for figure 8.9. The ***right*** panel shows the analytically predicted values using equation (8.36). Both matrices have had the diagonal (variances in figure 8.9) removed. While there is clearly a large discrepancy between the two figures, it is also clear that they share some features.

8.6 Summary

In this chapter, I have presented a novel approach to reducing the dimension of dynamics on large scale networks. By sourcing the variables of the full equation with the reduced variables, I have shown that it is possible for a variety of networks to decouple the original variables from the specific activity of their neighbors, and approximate the dynamics of the full system with a relatively small number of variables representing macroscopic dynamics together with the local topology (e.g. the degree sequence). At the very least, this interplay between local network topology and low dimensional dynamics is elegant. It is conceivable that this approach could lead to a framework within which one could analyze the effects of network structures at different scales on dynamical phenomena.

Chapter 9

Addendum - the Langevin equation approximation for the PSM

In this chapter, I will expand upon the path integral formulation of the Poisson spiking model (PSM), and derive a Langevin equation which approximates the PSM. The work in this chapter is ongoing, and was done in collaboration with Michael Buice.

9.1 Deriving the Langevin equation

To derive a Langevin equation corresponding to the PSM, we will construct Gaussian processes which best approximate the noise in the PSM. One of these processes will correspond to the Poisson spiking in the PSM, while another will reflect the quenched noise in the structure of the network. The method will be to expand the the action for the PSM to second order in the response (tilde) variables. Then by comparison with the action of a Gaussian process, and imposing some consistency conditions we will derive a colored noise process corresponding to the PSM. The statistics of the resulting Gaussian process will depend on the first and second order statistics of the network, i.e. p, α_{chn}, α_{cnv}, and α_{div}, from chapter 2. The derivation will employ the machinery of path integrals reviewed in chapter 7.

One of the inspirations for the approach in this chapter is the paper by Sompolisnky et al, [46] mentioned in the previous chapter, example 8.3.2. By applying a similar technique as the one below, they were able to provide a criterion for a system with quenched noise to demonstrate chaos. One of the future directions of the material in this chapter could be to determine if, or how second order structure of a network impacts the onset of chaos. Before that, however, there is still work that needs to be done to clarify the meaning and applicability of the Langevin equation we find. For instance, we will see that our derivation yields a colored noise process which has infinite memory, and this may limit the scope of its applicability.

Technically, the equation I will derive is a dynamical system driven by colored Gaussian noise [28]. We have been referring to such a system as a *Langevin equation*, though some authors reserve this term for Markov processes [21]. Generally systems driven by colored noise processes are not Markov. On the other hand, a large class of such systems may be embedded into larger systems which are Markov [28]. In any case, I will continue to refer to the system derived in this chapter as a *Langevin equation*.

In chapter 6, I presented two ways of looking at the PSM, one in terms of s variables and the other in terms of g variables. The difference between these two formulations is that s^i records the spikes of the i^{th} node, while g^i records spikes from the inputs to the i^{th} node. The previous two chapters have focused on the s variables. In this chapter we will focus on the g variables. Given an adjacency matrix of a network, the PSM can be written in terms of the g's as

$$\tau \dot{g}^i = -g^i + \kappa \sum_j W_j^i \sum_k \delta(t - t_k^j), \tag{9.1}$$

where the spike times $\{t_k^j\}$ are generated according to an inhomogeneous Poisson process with instantaneous rate $\phi(\kappa g^j(t) + b^j)$. In terms of s, we have that

$$g^i = \sum_k W_k^i s^k.$$

The reason we will focus on the g's in this chapter is that it is only for those variables that a central limit theorem *could* hold with respect to the limits we will

consider here. The assumption that I'm making is that if $\kappa = \mathcal{O}(1/N)$, $N \to \infty$, and p fixed then the noise term in (9.1) is well approximated by a Gaussian process. The noise (jumps) in the s variables don't exhibit a central limit theorem with respect to N/κ, since the size of the jumps doesn't change with these parameters. In our formulation of the PSM, τ *does* effect the amplitude of the spikes, but we will not be considering limits with respect to τ here.

In section 7.2, I derived an action corresponding to a stochastic differential equation (dynamical system driven by white noise). Below we will need the action corresponding to a more general equation:

$$\mathrm{d}X = F(X(t), t) + \mathrm{d}\xi(t) + \mathrm{d}\eta(t), \tag{9.2}$$

where the $\mathrm{d}\xi(t)$ and $\mathrm{d}\eta(t)$ correspond to Gaussian processes with $\langle \mathrm{d}\xi(t) \rangle = \langle \mathrm{d}\eta(t) \rangle = 0$, and

$$\left\langle \mathrm{d}\xi(t)\mathrm{d}\xi(t') \right\rangle = \Sigma(t)\delta(t - t'),$$

and

$$\left\langle \mathrm{d}\xi(t)\mathrm{d}\xi(t') \right\rangle = K(t, t').$$

In general, $\mathrm{d}\xi(t)$, and $\mathrm{d}\eta(t)$ cannot be combined into a single term, because they may scale differently with different parameterizations of time. In the Langevin equation corresponding to the PSM we will find there are two noise terms to consider. The first one will be associated with the Poisson spiking behavior the PSM, and will be delta correlated in time. The second noise term will be a colored noise term corresponding to the quenched noise of the network. The action corresponding to (9.2) is [10]

$$\begin{aligned} S[\tilde{X}, X] = -\int_{t_0}^{\infty} \left\{ \mathrm{d}t\tilde{X}(t)(\dot{X}(t) - F(X(t), t)) + \frac{1}{2}\tilde{X}(t)\Sigma(t)\tilde{X}(t)^T \right\} \\ - \frac{1}{2}\int_{t_0}^{\infty} \mathrm{d}t \int_{t_0}^{\infty} \mathrm{d}t' \tilde{X}(t)K(t, t')\tilde{X}(t')^T. \end{aligned} \tag{9.3}$$

In deriving a Langevin equation corresponding to the PSM, I will use several facts about the relationship between (9.2), and the corresponding action (9.3):

1. the term which is linear in the response variable $\tilde{X}$ corresponds to the deterministic part of the Langevin equation

2. the term which is quadratic in $\tilde{X}$ at equal times corresponds to the Gaussian process $\mathrm{d}\xi$, which is δ-correlated in time

3. the term which is quadratic in $\tilde{X}$ at different times corresponds to the colored Gaussian process $\mathrm{d}\eta$.

I'll now present the action for the PSM in terms of the variables g^i. Here we have a N nodes connected by the adjacency matrix W. As noted above, the dynamics of the system are given by

$$\tau \dot{g}^i = -g^i + \kappa \sum_j W_j^i \sum_k \delta(t - t_k^j), \tag{9.4}$$

where the spike times $\{t_k^j\}$ of node j are produced by an inhomogeneous Poisson process with instantaneous rate $\phi(\kappa g^j + b^j)$. The function ϕ is typically taken to be sigmoidal. For this system the Euler-Maruyama steps are

$$g_{n+1}^i = g_n^i - \frac{h}{\tau} g_n^i + \frac{\kappa}{\tau} \sum_j W_j^i \eta_n^j + g(t_0)\delta_{n,0}, \tag{9.5}$$

where $\eta_n^j \sim \mathrm{Poiss}(h\phi(g_n^j + b_n^j))$. Following the same steps as in the case of an SDE with Gaussian noise in section 7.2 we arrive at the action (Note, W is taken as fixed so far.):

$$S_W[\tilde{g}, g] = \sum_i \int_{t_0}^{\infty} \mathrm{d}t\, \tilde{g}_i \left(\dot{g}^i + \frac{g^i}{\tau} \right) - \sum_j \left(\exp\left(\frac{\kappa}{\tau} \sum_i \tilde{g}_i W_j^i \right) - 1 \right) \phi(\kappa g^j + b^j) - S_{\mathrm{init}}. \tag{9.6}$$

9.2 Marginalization of the PSM over an ensemble of networks & inference of the Langevin equation

Up to this point we have assumed that the network is fixed. We now suppose that the network has been drawn from a distribution, and want to marginalize the path

density over a distribution of networks. In the case that the system is self-averaging the resulting equations could give us insight into the behavior of individual systems, or *typical* systems.

To determine the effects of network structure on the noise of the PSM we want to marginalize the path density

$$P_W[\tilde{g}, g] = \exp\left(-S_W[\tilde{g}, g]\right)$$

over the distribution of the network W. Thus we want to perform the integral (over a finite dimensional space of an infinite dimensional functional)

$$\int \mathrm{d}W P_W[\tilde{g}, g] = \int \mathrm{d}W \exp\left(-S_W[\tilde{g}, g] + \log(P[W])\right).$$

Keeping those parts of the action which have a W, the integral we care about is

$$\int \mathrm{d}W \exp\left(\int_{t_0}^{\infty} \mathrm{d}t \sum_j \left(\exp\left(\frac{\kappa}{\tau}\sum_i \tilde{g}^i W_j^i\right) - 1\right)\phi(\kappa g^j + b^j)\right) P[W].$$

Moreover, since ultimately we want to derive a Langevin equation representing the marginalized dynamics, we need only retain terms which are linear and quadratic in the $\tilde{g}$ response variables. Thus we have

$$\int \mathrm{d}W \exp\left(\int_{t_0}^{\infty} \mathrm{d}t \sum_j \left(\frac{\kappa}{\tau}\sum_i \tilde{g}^i W_j^i + \frac{1}{2}\left(\frac{\kappa}{\tau}\sum_i \tilde{g}^i W_j^i\right)^2\right)\phi(\kappa g^j + b^j)\right) P[W]. \tag{9.7}$$

The details of the calculation of the integral in (9.7) are in section 9.3, below. The end result is the action

$$\begin{aligned} S[\tilde{g}, g] = \sum_i \int_{t_0}^{\infty} \mathrm{d}t \tilde{g}^i \left(\dot{g}^i + \frac{g^i}{\tau}\right) \\ - \frac{\kappa}{\tau}\tilde{g}^T \langle W_j^i \rangle \phi(\kappa g^j + b^j) - \frac{\kappa^2}{2\tau^2}\tilde{g}^i \tilde{g}^{i'} \langle W_j^i W_j^{i'} \rangle \phi(\kappa g^j + b^j) \\ - \frac{\kappa^2}{2\tau^2}\int_{t_0}^{\infty} \mathrm{d}t \int_{t_0}^{\infty} \mathrm{d}t' \tilde{g}^i(t) \tilde{g}^{i'}(t') \langle W_j^i W_{j'}^{i'} \rangle_c \phi(\kappa g^j(t) + b^j) \phi(\kappa g^{j'}(t') + b^{j'}) \end{aligned} \tag{9.8}$$

In (9.8) the angled brackets indicate expectations over $P[W]$, and *sub-c* means

cumulant (with respect to $P[W]$, when W's are in the brackets). For the above to apply to an individual system in which the network is frozen (and sampled from $P[W]$) one would have to have that the systems are self averaging or ergodic.

Comparing the marginalized action (9.8) with the action (9.3), we propose that a Langevin equation, the solutions of which produce sample paths from the marginalized path density, is

$$\dot{g}^i = -\frac{1}{\tau} g^i + \frac{\kappa}{\tau} \langle W_j^i \rangle \phi(\kappa g^j + b^j) + \mathrm{d}\xi^i + \mathrm{d}\eta^i, \tag{9.9}$$

where $\mathrm{d}\xi$ is a Gaussian noise term which is δ-correlated in time and corresponds to the spiking (this part of the noise appears in [3]), and $\mathrm{d}\eta$ is a colored Gaussian noise term resulting from the quenched/frozen noise of the network. In particular, we take both $\mathrm{d}\xi$ and $\mathrm{d}\eta$ to have mean zero, and

$$\langle \mathrm{d}\xi^i(t) \mathrm{d}\xi^{i'}(t') \rangle = \frac{\kappa^2}{\tau^2} \langle W_j^i W_j^{i'} \rangle \langle \phi(\kappa g^j(t) + b^j) \rangle \delta(t - t'), \tag{9.10}$$

$$\langle \mathrm{d}\eta^i(t) \mathrm{d}\eta^{i'}(t') \rangle = \frac{\kappa^2}{\tau^2} \langle W_j^i W_{j'}^{i'} \rangle_c \langle \phi(\kappa g^j(t) + b^j) \phi(\kappa g^{j'}(t') + b^j) \rangle. \tag{9.11}$$

Because $\mathrm{d}\eta$, and $\mathrm{d}\xi$ are noise at different scales of times, and to avoid subtle interactions between these terms I will assume they are independent,

$$\langle \mathrm{d}\eta^i(t) \mathrm{d}\xi^{i'}(t') \rangle = 0. \tag{9.12}$$

In the above equations, expectations of the ϕ's are with respect to the noise terms and may be computed self consistently, perhaps by a method similar to [46].

Notice that the expectation of the ϕ's in equation (9.11) is a full second moment, and so the colored noise of the Langevin equation corresponding to the PSM has *infinite memory*. In terms of the derivation, this is due to the fact that the noise in the network is frozen, but it is not clear what this would imply for the Langevin equation.

For example, consider the SDE

$$\mathrm{d}x(t) = \mathrm{d}\xi(t), \tag{9.13}$$

where $\langle \mathrm{d}\xi(t) \rangle = 0$, and $\langle \mathrm{d}\xi(t)\mathrm{d}\xi(t') \rangle = c > 0$ for all t, and t'. This implies that $\mathrm{d}\xi(t)$ is constant, since the rank of the covariance is 1. Therefore any solution $x(t)$ of (9.13) will drift linearly away from its initial condition. If the infinite memory in $\mathrm{d}\xi$ in the Langevin equation leads to a linear drift term, then its utility is dubious, since the dynamics of the PSM are bounded both above and below for any ϕ which is bounded. At this time, the implications/validity of the Langevin equation remain open to further exploration.

9.3 Performing the integral over the ensemble of networks

In this appendix I will detail the steps of going from (9.7) to (9.8). The key to this calculation relies on two things: 1) we need only retain terms to second order in the response $\tilde{g}$ variables, and 2) the relationship between the moment generating function and the cumulant generating function. The part of the PSM action which includes W terms is, as in (9.7),

$$\begin{aligned} &\int \mathrm{d}W \exp\left(\int_{t_0}^{\infty} \mathrm{d}t \left(\tilde{g}_i W_j^i + \frac{1}{2} \left(\tilde{g}_i W_j^i \right)^2 \right) \phi(g^j + b^j) \right) P[W] \\ &= \int \mathrm{d}W \exp\left(\left[\int_{t_0}^{\infty} \mathrm{d}t \tilde{g}_i(t) \phi(g^j(t) + b^j) \right] W_j^i \right. \\ &\qquad \left. + \frac{1}{2} \left[\int_{t_0}^{\infty} \mathrm{d}t \tilde{g}_i(t) \tilde{g}_{i'}(t) \phi(g^j(t) + b^j) \right] W_j^i W_j^{i'} \right) P[W]. \end{aligned} \tag{9.14}$$

I am following the Einstein notation convention here. That is, any index which appears both as a superscript and a subscript is summed over. Letting

$$\alpha_i^j = \int_{t_0}^{\infty} \mathrm{d}t \tilde{g}_i(t) \phi(g^j(t) + b^j),$$

and

$$\beta_{ii'}^j = \int_{t_0}^{\infty} \mathrm{d}t \tilde{g}_i(t) \tilde{g}_{i'}(t) \phi(g^j(t) + b^j),$$

we can express (9.14) more compactly as

$$\int \mathrm{d}W \exp\left(\alpha_i^j W_j^i + \frac{1}{2} \beta_{ii'}^j W_j^i W_j^{i'} \right) P[W].$$

Our goal is to perform this integral keeping only terms linear or quadratic in the response variables $\tilde{g}$. Notice that there is one response variable in α_i^j and two response variables in $\beta_{ii'}^j$. We proceed as

$$
\begin{aligned}
&\int \mathrm{d}W \exp\left(\alpha_i^j W_j^i + \frac{1}{2}\beta_{ii'}^j W_j^i W_j^{i'}\right) P[W] \\
&= \int \left(1 + \alpha_i^j W_j^i + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} W_j^i W_{j'}^{i'} + \ldots\right)\left(1 + \frac{1}{2}\beta_{ii'}^j W_j^i W_j^{i'} + \ldots\right) P[W]\mathrm{d}W \\
&= \int \left(1 + \alpha_i^j W_j^i + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} W_j^i W_{j'}^{i'} + \ldots\right) P[W]\mathrm{d}W + \int \frac{1}{2}\beta_{ii'}^j W_j^i W_j^{i'} \left(1 + \ldots\right) P[W]\mathrm{d}W + \ldots \\
&= 1 + \alpha_i^j \langle W_j^i\rangle + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \langle W_j^i W_{j'}^{i'}\rangle + \frac{1}{2}\beta_{ii'}^j \langle W_j^i W_j^{i'}\rangle + \ldots \\
&= 1 + \alpha_i^j \langle W_j^i\rangle + \frac{1}{2}\beta_{ii'}^j \langle W_j^i W_j^{i'}\rangle + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \left(\langle W_j^i W_{j'}^{i'}\rangle - \langle W_j^i\rangle\langle W_{j'}^{i'}\rangle\right) + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \langle W_j^i\rangle\langle W_{j'}^{i'}\rangle + \ldots \\
&\approx 1 + \alpha_i^j \langle W_j^i\rangle + \frac{1}{2}\beta_{ii'}^j \langle W_j^i W_j^{i'}\rangle + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \langle W_j^i W_{j'}^{i'}\rangle_c \\
&\qquad\qquad + \frac{1}{2}\left(\alpha_i^j \langle W_j^i\rangle + \frac{1}{2}\beta_{ii'}^j \langle W_j^i W_j^{i'}\rangle + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \langle W_j^i W_{j'}^{i'}\rangle_c\right)^2 + \ldots \\
&\approx \exp\left(\alpha_i^j \langle W_j^i\rangle + \frac{1}{2}\beta_{ii'}^j \langle W_j^i W_j^{i'}\rangle + \frac{1}{2}\alpha_i^j \alpha_{i'}^{j'} \langle W_j^i W_{j'}^{i'}\rangle_c\right).
\end{aligned}
\tag{9.15}
$$

Applying the definitions of α and β and recombining this with the action (9.6) completes the calculation.

Bibliography

[1] Paul Blanchard, Robert L Devaney, Antonio Garijo, Sebastian M Marotta, and Elizabeth D Russell. Rabbits, basilicas, and other julia sets wrapped in sierpinski carpets. *to appear*, 2009.

[2] John Brewer. Kronecker products and matrix calculus in system theory. *IEEE Transactions on circuits and systems*, 25(9):772–781, 1978.

[3] Michael A Buice. Network of poisson neurons. In preperation, November 2012.

[4] Michael A Buice, Jack D Cowan, and Carson C Chow. Systematic fluctuation expansion for neural network activity equations. *Neural computation*, 22(2):377–426, 2010.

[5] Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, 2009.

[6] Steve Butler. Using discrepancy to control singular values for nonnegative matrices. *Linear algebra and its applications*, 419(2):486–493, 2006.

[7] Steven Butler. Relating singular values and discrepancy of weighted directed graphs. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1112–1116. ACM, 2006.

[8] Steven Kay Butler. *Eigenvalues and structures of graphs*. ProQuest, 2008.

[9] Masud Chaichian and A Demichev. *Path Integrals in Physics, Vol. 1: Stochastic Processes and Quantum Mechanics*. IOP, Bristol, UK, 2001.

[10] C Chow and M Buice. Path integral methods for stochastic differential equations. *arXiv preprint arXiv:1009.5966*, 2010.

[11] Fan RK Chung and Linyuan Lu. *Complex graphs and networks*, volume 107. American mathematical society Providence, 2006.

[12] David R Cox. Some statistical methods connected with series of events. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 129–164, 1955.

[13] Peter Dayan and Laurence Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems (Computational Neuroscience)*. The MIT Press, 1 edition, 2005.

[14] Ronald Dickman and Ronaldo Vidigal. Path integrals and perturbation theory for stochastic processes. *Brazilian Journal of physics*, 33(1):73–93, 2003.

[15] Masao Doi. Stochastic theory of diffusion-controlled reaction. *Journal of Physics A: Mathematical and General*, 9(9):1479, 1976.

[16] Victor M Eguiluz, Dante R Chialvo, Guillermo A Cecchi, Marwan Baliki, and A Vania Apkarian. Scale-free brain functional networks. *Physical review letters*, 94(1):018102, 2005.

[17] Paul Erdős and Alfréd Rényi. On random graphs. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[18] Lawrence C Evans. An introduction to stochastic differential equations version 1.2. *Department of Mathematics UC Berkeley, in internet*, 2001.

[19] Richard Phillips Feynman and Albert Roach Hibbs. *Quantum mechanics and path integrals*. McGraw-Hill, 1965.

[20] Jacob G Foster, David V Foster, Peter Grassberger, and Maya Paczuski. Edge direction and the structure of networks. *Proceedings of the National Academy of Sciences*, 107(24):10815–10820, 2010.

[21] CW Gardiner. Handbook of stochastic methods for physics, chemistry and the natural sciences. *Applied Optics*, 25:3145, 1986.

[22] Wulfram Gerstner and Werner Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 1 edition, 2002.

[23] Edgar N Gilbert. Random graphs. *The Annals of Mathematical Statistics*, pages 1141–1144, 1959.

[24] Anna Goldenberg, Alice X Zheng, Stephen E Fienberg, and Edoardo M Airoldi. A survey of statistical network models. *Foundations and Trends® in Machine Learning*, 2(2):129–233, 2010.

[25] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[26] Martin Golubitsky and Ian Stewart. Nonlinear dynamics of networks: the groupoid formalism. *Bulletin of the american mathematical society*, 43(3):305–364, 2006.

[27] John A Gubner. A brief history of shot noise.

[28] Peter Hanggi and Peter Jung. Colored noise in dynamical systems. *Advances in chemical physics*, 89:239–326, 1995.

[29] Yu Hu, James Trousdale, Krěsimir Josić, and Eric Shea-Brown. Motif statistics and spike correlations in neuronal networks. *BMC Neuroscience*, 13:1–2, 2012.

[30] Leon Isserlis. On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, pages 134–139, 1918.

[31] Dan Kalman. A singularly valuable decomposition: the svd of a matrix. In *College Math Journal*. Citeseer, 1996.

[32] Peter E Kloeden and Eckhard Platen. *Numerical solution of stochastic differential equations*, volume 23. Springer, 1992.

[33] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: An approach to modeling networks. *The Journal of Machine Learning Research*, 11:985–1042, 2010.

[34] Jurij Leskovec, Deepayan Chakrabarti, Jon Kleinberg, and Christos Faloutsos. Realistic, mathematically tractable graph generation and evolution, using kronecker multiplication. In *Knowledge Discovery in Databases: PKDD 2005*, pages 133–145. Springer, 2005.

[35] Pradipta Mitra. Entrywise bounds for eigenvectors of random graphs. *the electronic journal of combinatorics*, 16(1):R131, 2009.

[36] John A Nelder and RJ Baker. *Generalized linear models*. Wiley Online Library, 1972.

[37] Theoden I Netoff, Robert Clewley, Scott Arno, Tara Keck, and John A White. Epilepsy in small-world networks. *The Journal of Neuroscience*, 24(37):8075–8083, 2004.

[38] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[39] Athanasios Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill Companies, 3rd edition, 1991.

[40] L Peliti. Path integral approach to birth-death processes on a lattice. *Journal de Physique*, 46(9):1469–1483, 1985.

[41] Rodrigo Perin, Thomas K Berger, and Henry Markram. A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences*, 108(13):5419–5424, 2011.

[42] Juan G Restrepo, Edward Ott, and Brian R Hunt. Approximating the largest eigenvalue of network adjacency matrices. *arXiv preprint arXiv:0705.4503*, 2007.

[43] Sheldon M Ross. *Stochastic processes*. Wiley, New York, 1996.

[44] Gordon M Shepherd et al. *The synaptic organization of the brain*, volume 3. Oxford University Press New York, 2004.

[45] Sean L Simpson, Satoru Hayasaka, and Paul J Laurienti. Exponential random graph modeling for complex brain networks. *PLoS One*, 6(5):e20039, 2011.

[46] Haim Sompolinsky, A Crisanti, and HJ Sommers. Chaos in random neural networks. *Physical Review Letters*, 61(3):259, 1988.

[47] Sen Song, Per Jesper Sjöström, Markus Reigl, Sacha Nelson, and Dmitri B Chklovskii. Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS biology*, 3(3):e68, 2005.

[48] Olaf Sporns. Networks of the brain: Quantitative analysis and modeling. *Analysis and Function of Large-Scale Brain Networks*, page 7, 2010.

[49] Olaf Sporns. *Networks of the Brain*. MIT press, 2011.

[50] Olaf Sporns, Giulio Tononi, and Rolf Kötter. The human connectome: a structural description of the human brain. *PLoS computational biology*, 1(4):e42, 2005.

[51] Arthur W Toga, Kristi A Clark, Paul M Thompson, David W Shattuck, and John Darrell Van Horn. Mapping the human connectome. *Neurosurgery*, 71(1):1, 2012.

[52] James Trousdale, Yu Hu, Eric Shea-Brown, and Krešimir Josić. Impact of network structure and cellular response on spike time correlations. *PLoS computational biology*, 8(3):e1002408, 2012.

[53] Carl van Vreeswijk and Haim Sompolinsky. Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science*, 274(5293):1724–1726, 1996.

[54] Thomas Wanner. An introduction to stochastic differential equations version 1.2, January 2013.

[55] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[56] Horacio S Wio, P Colet, M San Miguel, L Pesquera, and MA Rodriguez. Path-integral formulation for stochastic processes driven by colored noise. *Physical Review A*, 40(12):7312, 1989.

[57] Liqiong Zhao. *Synchronization on Second Order Netowrks.* PhD thesis, University of Minnesota, October 2011.

[58] Liqiong Zhao, Bryce Beverlin, Tay Netoff, and Duane Quinn Nykamp. Synchronization from second order network connectivity statistics. *Frontiers in Computational Neuroscience*, 5(0), 2011.

[59] Jean Zinn-Justin. Quantum field theory and critical phenomena. Technical report, 2002.

Appendix A

Path integral toy problems

The following is a collection of five path integral problems which may be useful for someone else learning path integrals. I will assume familiarity with Feynman diagrams and recommend Buice and Chow's review which can be found online [10].

A.1 Restricted sigmoid no Poisson noise

By assuming the sigmoid ϕ is of a special type, we can simplify the equations (at a cost of doubling the dimension of the system). I'll assume that ϕ satisfies a differential equation of the form

$$\phi' = D(\phi) = (\phi - r_1)(r_2 - \phi) = -r_1 r_2 + (r_2 - r_1)\phi - \phi^2, \tag{A.1}$$

with $r_1 < r_2$. Thus, as long as $r_1 < \phi(0) < r_2$, ϕ is a sigmoid with asymptotes r_1 and r_2. Now let $Y^i = \phi(X^i + b^i)$. Then we can write our original SDE as

$$\mathrm{d}X^i = -\frac{1}{\tau}X^i \mathrm{d}t + \frac{\alpha}{\tau}W^i_j \mathrm{d}\pi^j \tag{A.2}$$

$$\mathrm{d}Y^i = \frac{\gamma}{r}(r - Y^i)Y^i(-\frac{1}{\tau}X^i \mathrm{d}t + \frac{\alpha}{\tau}W^i_j \mathrm{d}\pi^j). \tag{A.3}$$

Moreover, this equation is a weak coupling approximation of the PSM, in which the jump size of the Y's (the function multiplying the spikes) has been approximated by the derivative of the sigmoid. An approximation that holds when $\alpha << \tau$. It is nice the initial condition of Y^i corresponds to the constant b^i.

Consider the deterministic system (setting $\tau = 1$)

$$\dot{X}^i = -X^i + \alpha W^i_j Y^j \tag{A.4}$$

$$\dot{Y}^i = D(Y^i)(-X^i + \alpha W^i_j Y^j), \tag{A.5}$$

where D is defined as in (A.1). Let $D_0 = -r_1 r_2$. After marginalizing over the distribution of W, and truncating the action at second order in the response variables we have the action

$$S[\tilde{X}, \tilde{Y}, X, Y] = S_F[\tilde{X}, \tilde{Y}, X, Y] + S_I[\tilde{X}, \tilde{Y}, X, Y],$$

where

$$S_F[\tilde{X}, \tilde{Y}, X, Y] = \int_{t_0}^{\infty} \tilde{X}_i \left(\dot{X}^i + X^i - \alpha \langle W^i_j \rangle Y^j \right) + \tilde{Y}_i \left(\dot{Y}^i + D_0 X^i - \alpha D_0 \langle W^i_j \rangle Y^j \right),$$

and

$$\begin{aligned} S_I[\tilde{X}, \tilde{Y}, X, Y] = &\int_{t_0}^{\infty} \tilde{Y}_i (D(Y^i) - D_0) X^i - \alpha \tilde{Y}_i (D(Y^i) - D_0) \langle W^i_j \rangle Y^j \\ &- \frac{\alpha^2}{2} \langle W^{i'}_{j'} W^i_j \rangle_c \left(\int_{t_0}^{\infty} \mathrm{d}t' \tilde{X}_{i'} Y^{j'} + \tilde{Y}_{i'} D(Y^{i'}) Y^{j'} \right) \left(\int_{t_0}^{\infty} \mathrm{d}t \tilde{X}_i Y^j + \tilde{Y}_i D(Y^i) Y^j \right) \end{aligned}$$

The free part of the action has inverse propagators:

$$\Delta^{-1}_{\tilde{X}X}(t', t)^i_j = \delta^i_j \delta(t' - t) \left(\frac{\mathrm{d}}{\mathrm{d}t} + 1 \right) \tag{A.6}$$

$$\Delta^{-1}_{\tilde{X}Y}(t', t)^i_j = \delta(t' - t) \alpha \langle W^i_j \rangle \tag{A.7}$$

$$\Delta^{-1}_{\tilde{Y}Y}(t', t)^i_j = \delta(t' - t) \left(\delta^i_j \frac{\mathrm{d}}{\mathrm{d}t} - \alpha D_0 \langle W^i_j \rangle \right) \tag{A.8}$$

$$\Delta^{-1}_{\tilde{Y}X}(t', t)^i_j = \delta(t' - t) \delta^i_j D_0. \tag{A.9}$$

The free propagators satisfy

$$\int_{t_0}^{\infty} \mathrm{d}s \begin{pmatrix} \Delta^{-1}_{\tilde{X}X}(t',s)^i_k & \Delta^{-1}_{\tilde{X}Y}(t',s)^i_k \\ \Delta^{-1}_{\tilde{Y}X}(t',s)^i_k & \Delta^{-1}_{\tilde{Y}Y}(t',s)^i_k \end{pmatrix} \begin{pmatrix} \Delta_{X\tilde{X}}(s,t)^k_j & \Delta_{X\tilde{Y}}(s,t)^k_j \\ \Delta_{Y\tilde{X}}(s,t)^k_j & \Delta_{Y\tilde{Y}}(s,t)^k_j \end{pmatrix} \tag{A.10}$$
$$= \begin{pmatrix} \delta^i_j \delta(t'-t) & 0 \\ 0 & \delta^i_j \delta(t'-t) \end{pmatrix}.$$

As a system of equations, we find

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} \Delta_{X\tilde{X}}(t',t)^i_j \\ \Delta_{X\tilde{Y}}(t',t)^i_j \\ \Delta_{Y\tilde{X}}(t',t)^i_j \\ \Delta_{Y\tilde{Y}}(t',t)^i_j \end{pmatrix}$$

$$= \begin{pmatrix} -\delta^i_k & 0 & \alpha\langle W^k_j\rangle & 0 \\ 0 & -\delta^i_k & 0 & \alpha\langle W^k_j\rangle \\ -D_0\delta^i_k & 0 & \alpha D_0\langle W^k_j\rangle & 0 \\ 0 & -D_0\delta^i_k & 0 & D_0\alpha\langle W^k_j\rangle \end{pmatrix} \begin{pmatrix} \Delta_{X\tilde{X}}(t',t)^k_j \\ \Delta_{X\tilde{Y}}(t',t)^k_j \\ \Delta_{Y\tilde{X}}(t',t)^k_j \\ \Delta_{Y\tilde{Y}}(t',t)^k_j \end{pmatrix} + \begin{pmatrix} \delta^i_j\delta(t'-t) \\ 0 \\ 0 \\ \delta^i_j\delta(t'-t) \end{pmatrix}. \tag{A.11}$$

Following the Ito convention we take $\Delta(t,t) = 0$ for each propagator. Therefore the

propagators are given by

$$\Delta(t',t) = \int_t^{t'} \mathrm{d}s \exp((t'-s)A) \begin{pmatrix} \delta^i_j \delta(t'-s) \\ 0 \\ 0 \\ \delta^i_j \delta(t'-s) \end{pmatrix}, \tag{A.12}$$

where A is the coefficient matrix in (A.11). Exponentiating $(t'-s)A$ yields

$$\exp((t'-s)A) = \begin{pmatrix} \mathcal{E}\mathcal{W}^{-1} + I & 0 & -\alpha\mathcal{E}\mathcal{W}^{-1}\langle W\rangle & 0 \\ 0 & \mathcal{E}\mathcal{W}^{-1} + I & 0 & -\alpha\mathcal{E}\mathcal{W}^{-1}\langle W\rangle \\ D_0\mathcal{E}\mathcal{W}^{-1} & 0 & -\alpha D_0\mathcal{E}\mathcal{W}^{-1}\langle W\rangle + I & 0 \\ 0 & D_0\mathcal{E}\mathcal{W}^{-1} & 0 & -\alpha D_0\mathcal{E}\mathcal{W}^{-1}\langle W\rangle + I \end{pmatrix},$$

where

$$\mathcal{W} = I - \alpha D_0 \langle W\rangle,$$

and

$$\mathcal{E} = \exp(-(t'-t)\mathcal{W}) - I.$$

Putting these together we find that the free propagators satisfy

$$\Delta_{X\tilde{X}}(t',t) = H(t'-t)(\exp(-(t'-t)\mathcal{W})\mathcal{W}^{-1} + I) \tag{A.13}$$

$$\Delta_{X\tilde{Y}}(t',t) = H(t'-t)(-\alpha \exp(-(t'-t)\mathcal{W})\mathcal{W}^{-1}\langle W\rangle) \tag{A.14}$$

$$\Delta_{Y\tilde{X}}(t',t) = H(t'-t)(D_0 \exp(-(t'-t)\mathcal{W})\mathcal{W}^{-1}) \tag{A.15}$$

$$\Delta_{Y\tilde{Y}}(t',t) = H(t'-t)(-\alpha D_0 \exp(-(t'-t)\mathcal{W})\mathcal{W}^{-1}\langle W\rangle + I). \tag{A.16}$$

Note: when $D_0 = 0$, i.e. 0 is an asymptote of the sigmoid, the propagators simplify. The terms of the action which are first order in the response variables determine

the equations for the tree level mean:

$$\dot{\bar{X}}^i = -\bar{X}^i + \alpha \langle W^i_j \rangle \bar{Y}^j \tag{A.17}$$

$$\dot{\bar{Y}}^i = D(\bar{Y}^i)(-\bar{X}^i + \alpha \langle W^i_j \rangle \bar{Y}^j). \tag{A.18}$$

Notice that the tree level means only depend on first order statistics of the network. The full mean depends on higher order network statistics.

Example A.1.1. Consider the very simple process

$$\mathrm{d}x = \alpha \mathrm{d}t$$

where $\alpha(t) = \alpha \sim \mathcal{N}(0, \sigma^2)$. Then $x(t) = \alpha t$, and $\langle x(t')x(t) \rangle_c = \sigma^2 t t'$. In particular this process has infinite memory, in that if two trials are correlated at some point in time, then they are correlated that way for all time.

Example A.1.2. Consider the process

$$\mathrm{d}x = (-ax + b)\mathrm{d}t + \mathrm{d}\xi$$

where $b(t) \equiv b \sim \mathcal{N}(\mu, \beta^2)$, and $\mathrm{d}\xi$ is delta-correlated Gaussian noise with variance σ^2. The action corresponding to this equation is

$$S[\tilde{x}, x] = \int \mathrm{d}t \tilde{x}(\dot{x} + ax) - \tilde{x}(0)x_0 - \mu \int \mathrm{d}t \tilde{x}(t) - \frac{\beta^2}{2} \int \mathrm{d}t \tilde{x}(t) \int \mathrm{d}t' \tilde{x}(t') - \frac{\sigma^2}{2} \int \mathrm{d}t \tilde{x}^2.$$

The free propagator is

$$\Delta(t', t) = H(t' - t) \exp(-a(t' - t)).$$

And the diagrams we get from the rest of the action are

$\tilde{x}(0)x_0 :$ $\mu\, \tilde{x}(t) :$  $\frac{\beta^2}{2}\tilde{x}(t)\tilde{x}(t') :$  $\frac{\sigma^2}{2} \int \mathrm{d}t \tilde{x}^2 :$  .

Since there are no ingoing edges in these diagrams it is fairly straightforward to

compute the mean and covariances for this process. We find

$$\langle x(t)\rangle = \Delta(t',0)x_0 + \mu\int_0^\infty \mathrm{d}s\Delta(t,s) = \exp(-at)x_0 + \frac{\mu}{a}(1-\exp(-at)).$$

and

$$\begin{aligned}\langle x(t')x(t)\rangle_c &= \sigma^2\int \mathrm{d}s\Delta(t',s)\Delta(t,s) + \beta^2\int \mathrm{d}s'\Delta(t',s')\int \mathrm{d}s\Delta(t,s)\\ &= \frac{\sigma^2}{2a}\left[\exp(-a(t'+t-2\min(t',t))) - \exp(-a(t'+t))\right]\\ &\qquad + \frac{\beta^2}{a^2}\left[\left(1-\exp(-at')\right)\left(1-\exp(-at)\right)\right].\end{aligned}$$

In particular, the second term of the covariance does not vanish. This means that there is no forgetting time.

Example A.1.3. Now say the b is fixed, and suppose a is random. So we're looking at

$$\mathrm{d}x = (-ax+b)\mathrm{d}t + \mathrm{d}\xi$$

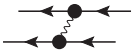
where $a(t)\equiv a\sim\mathcal{N}(\mu,\alpha^2)$, and $\mathrm{d}\xi$ is delta-correlated Gaussian noise with variance σ^2. The action corresponding to this equation is

$$S[\tilde{x},x] = \int \mathrm{d}t\tilde{x}(\dot{x}+\mu x) - \tilde{x}(0)x_0 - b\int \mathrm{d}t\tilde{x}(t) + \frac{\alpha^2}{2}\int \mathrm{d}t'\tilde{x}(t')x(t')\int \mathrm{d}t\tilde{x}(t)x(t) - \frac{\sigma^2}{2}\int \mathrm{d}t\tilde{x}^2.$$

The free propagator is

$$\Delta(t',t) = H(t'-t)\exp(-\mu(t'-t)).$$

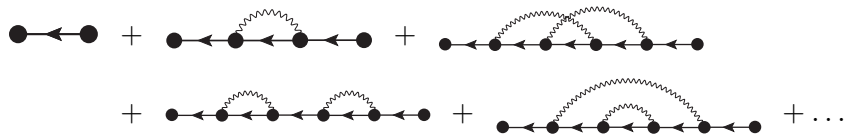
And the diagrams we get from the rest of the action are

$\tilde{x}(0)x_0$:  $b\,\tilde{x}$:  $-\frac{\alpha^2}{2}\tilde{x}(t')x(t')\tilde{x}(t)x(t)$: 

$\frac{\sigma^2}{2}\tilde{x}^2$:  .

We now have ingoing edges in the α^2 term, and that makes the computation of the mean slightly trickier. Expressing the mean in terms of diagrams (and combining

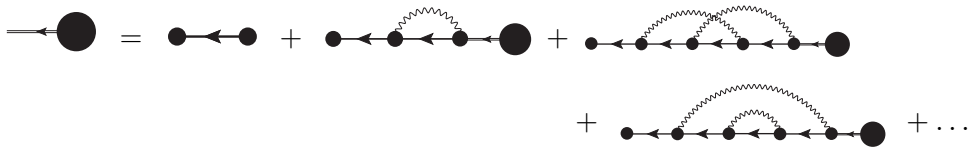
the diagrams for the b, and x_0 terms) yields

$\langle x(t)\rangle =$  $+\ldots$

Write the full mean as a single large vertex

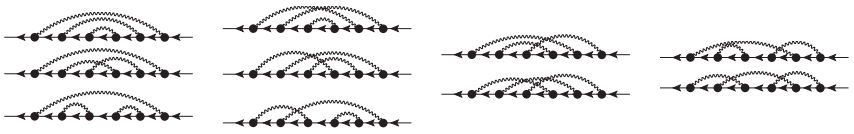
$\langle x(t)\rangle =$  .

Then the diagram expansion takes the form



(A.19)

Notice that one of the terms in the full expansion was absorbed into a diagram in the last expansion, but the other two 2-boson terms cannot be so easily absorbed. At the 3-boson order we get the following additional terms which can't be absorbed into lower order diagrams:



We need only keep track of the numbers of different types of diagrams because their integrals are all the same. Note

$$\int_{t_0}^{\infty} \Delta(t', s_1)\Delta(s_1, s_2)\cdots\Delta(s_n, t)\mathrm{d}s_1 \ldots \mathrm{d}s_n = \frac{1}{n!}(t'-t)^n \Delta(t', t),$$

and

$$\int_{t_0}^{\infty} \Delta(t', s_1)\Delta(s_1, s_2)\cdots\Delta(s_n, t)\mathrm{d}s_1 \ldots \mathrm{d}s_n \mathrm{d}t = \frac{1}{\mu^{n+1}}\left(1 - \Delta(t', t_0)\sum_{k=0}^{n}\frac{(\mu(t'-t_0))^k}{k!}\right).$$

Using this we can write down an equation for the full mean (taking $t_0 = 0$)

$$\langle x(t)\rangle = \sum_{n=0}^{\infty}\left(-\frac{\alpha^2}{2}\right)^n \frac{(2n)!}{n!}\left(\frac{x_0 t^{2n}}{(2n)!}\Delta(t,0) + \frac{b}{\mu^{2n+1}}\left(1-\Delta(t,0)\sum_{k=0}^{2n}\frac{(\mu t)^k}{k!}\right)\right),$$

which can be slightly simplified to

$$\langle x(t)\rangle = \Delta(t,0)x_0\exp\left(-\frac{\alpha^2 t^2}{2}\right) + \frac{b}{\mu}\sum_{n=0}^{\infty}\left(-\frac{\alpha^2}{2}\right)^n \frac{(2n)!}{n!}\left(\frac{1}{\mu^{2n}}\left(1-\Delta(t,0)\sum_{k=0}^{2n}\frac{(\mu t)^k}{k!}\right)\right).$$

The factorials are certainly concerning. Clearly, this series is not absolutely convergent, making the simplification of the last equation suspect. Finitely truncating this series provides functions which diverge off to infinity after some point. It may be worth pointing out that when I set $b = 0$ the tree level mean does better numerically than the full mean. That concerns me, but I don't see any mistakes, yet. The very nice thing would be if for some reason these quenched noise diagrams only apply as branches, and aren't able to feed into themselves... But that's not actually the case since it is also clear from the numerics that the mean of x definitely depends on α, which is not present in the tree level mean. On the other hand, the full mean, assuming my computation is correct is always dominated by the tree level mean, and from numerics lowering α pushes the true mean higher than the tree level. So even though there's no dependence on α in the tree level mean, it still appears to be more accurate than the full mean (all this for when $b = 0$).

Example A.1.4. Consider the following modified linear OU process:

$$\begin{aligned}\dot{x} &= ay + \xi\\ \dot{y} &= -ax + \zeta,\end{aligned}$$

where $a \sim \mathcal{N}(\mu, \alpha^2)$, $\xi \sim \mathcal{N}(0, \sigma^2)$, and $\zeta \sim \mathcal{N}(0, \eta^2)$.

Action:

$$\tilde{x}(\dot{x} - \mu y) + \tilde{y}(\dot{y} + \mu x) + \frac{\alpha^2}{2}[\tilde{y}x - \tilde{x}y]_{t'}[\tilde{y}x - \tilde{x}y]_t - \frac{\sigma^2}{2}[\tilde{x}^2]_t - \frac{\eta^2}{2}[\tilde{y}^2]_t$$

Inverse free propagators:

$$\Delta^{-1}_{\tilde{x}x}(t',t) = \delta(t'-t)\frac{\mathrm{d}}{\mathrm{d}t}$$
$$\Delta^{-1}_{\tilde{y}x}(t',t) = \delta(t'-t)\mu$$
$$\Delta^{-1}_{\tilde{x}y}(t',t) = -\delta(t'-t)\mu$$
$$\Delta^{-1}_{\tilde{y}y}(t',t) = \delta(t'-t)\frac{\mathrm{d}}{\mathrm{d}t}$$

And the free propagators:

$$\overline{\Delta}_{x\tilde{x}}(t',t) = H(t'-t)\cos(\mu(t'-t))$$
$$\overline{\Delta}_{y\tilde{x}}(t',t) = -H(t'-t)\sin(\mu(t'-t))$$
$$\overline{\Delta}_{x\tilde{y}}(t',t) = H(t'-t)\sin(\mu(t'-t))$$
$$\overline{\Delta}_{y\tilde{y}}(t',t) = H(t'-t)\cos(\mu(t'-t))$$

Since $\Delta_{y\tilde{y}} = \Delta_{x\tilde{x}}$, I'll use $\Delta_{z\tilde{z}}$ to denote either one. It is also worth knowing that all the propagators commute as in

$$\int \mathrm{d}s\Delta_{u\tilde{v}}(t',s)\Delta_{w\tilde{z}}(s,t) = \int \mathrm{d}s\Delta_{w\tilde{z}}(t',s)\Delta_{u\tilde{v}}(s,t).$$

Also note the relations

$$\int_{t_0}^{\infty} \mathrm{d}s\Delta_{z\tilde{z}}(t',s)\Delta_{z\tilde{z}}(s,t) = \frac{1}{2}(t'-t)\Delta_{z\tilde{z}}(t',t) + \frac{1}{2\mu}\Delta_{x\tilde{y}}(t',t),$$

The tree level means are the same as the free means, and the same as the mean in the case of a linear OU process.

$$\bar{x}(t) = \overline{\Delta}_{x\tilde{x}}(t,t_0)x_0 + \overline{\Delta}_{x\tilde{y}}(t,t_0)y_0$$
$$\bar{y}(t) = \overline{\Delta}_{y\tilde{x}}(t,t_0)x_0 + \overline{\Delta}_{y\tilde{y}}(t,t_0)y_0.$$

These are also the full mean in the absence of the quenched noise (i.e. $\alpha = 0$). In the absence of quenched noise we have the full covariances, as in the linear OU

process. So

$$\begin{aligned}&\langle x(t')x(t)\rangle_c\\&= \frac{1}{2\mu}\left(\mu(\eta^2+\sigma^2)\min(t',t)\cos(\mu(t'-t)) - (\eta^2-\sigma^2)\cos(\mu\max(t,t'))\sin(\mu\min(t,t'))\right),\end{aligned}$$

$$\begin{aligned}&\langle x(t')x(t)\rangle_c\\&= \frac{1}{2\mu}\left(\mu(\eta^2+\sigma^2)\min(t',t)\cos(\mu(t'-t)) + (\eta^2-\sigma^2)\cos(\mu\max(t,t'))\sin(\mu\min(t,t'))\right),\end{aligned}$$

and

$$\langle x(t')y(t)\rangle_c = \frac{1}{2\mu}\left(\mu(\eta^2+\sigma^2)\min(t',t)\sin(\mu(t'-t)) + (\eta^2-\sigma^2)\sin(\mu t')\sin(\mu t)\right).$$

To account for the quenched noise we have to include the other diagrams in the expansion. Using the commutation relation above, we can simplify the whole expansion. We find that the full propagators are just the free propagators multiplied by a decay function. That is, they each have the form

$$\Delta(t',t) = \exp(-\alpha^2(t'-t))\overline{\Delta}(t',t).$$

Thus, the full means are just

$$\begin{aligned}\bar{x}(t) &= \Delta_{x\tilde{x}}(t,t_0)x_0 + \Delta_{x\tilde{y}}(t,t_0)y_0\\ \bar{y}(t) &= \Delta_{y\tilde{x}}(t,t_0)x_0 + \Delta_{y\tilde{y}}(t,t_0)y_0.\end{aligned}$$

From here I can get exact solutions for the full covariances.

Example A.1.5. Before tackling the full network system, let's do one more low dimensional quenched noise system. Consider

$$\mathrm{d}x = (-x + a\phi(x))\mathrm{d}t + \mathrm{d}\xi,$$

where $a \sim \mathcal{N}(\mu, \alpha^2)$, and $\mathrm{d}\xi$ is white noise. We'll assume that the nonlinearity ϕ is a sigmoid of the form:

$$\phi(x) = \frac{\gamma}{1+\exp(-x+b)},$$

where b, and γ are fixed constants. This is a convenient choice of ϕ since it satisfies a simple differential equation:

$$\phi' = \frac{1}{\gamma}\phi(\gamma - \phi).$$

Thus setting $y = \phi(x)$, we can rewrite the system as a polynomial in its variables.

$$\begin{aligned} \mathrm{d}x &= (-x + ay)\mathrm{d}t + \mathrm{d}\xi \\ \mathrm{d}y &= \frac{1}{\gamma}y(\gamma - y)\left((\frac{1}{2} - x + (a - \frac{1}{\gamma})y)\mathrm{d}t + \mathrm{d}\xi\right) \end{aligned}$$

Notice that I have applied Ito's lemma in obtaining this system. The action corresponding action (obtained by marginalizing over the random variables in the system) is $S[\tilde{x}, \tilde{y}, x, y] = S_F[\tilde{x}, \tilde{y}, x, y] + S_I[\tilde{x}, \tilde{y}, x, y]$, where

$$S_F[\tilde{x}, \tilde{y}, x, y] = \tilde{x}(\dot{x} + x - \mu y) + \tilde{y}(\dot{y} - \frac{1}{2}y),$$

and, letting $z = \frac{1}{\gamma}y(\gamma - y)$,

$$S_I[\tilde{x}, \tilde{y}, x, y] = \tilde{y}(\frac{1}{2\gamma}y^2 + xz - (\mu - \frac{1}{\gamma})zy)) - \frac{\alpha^2}{2}[\tilde{x}y + \tilde{y}yz]_{t'}[\tilde{x}y + \tilde{y}yz]_t - \frac{\sigma^2}{2}[(\tilde{x} + \tilde{y}z)^2]_{t'}.$$

The tree level mean satisfies the equation implied by the terms which are order one in the response variables. Namely,

$$\begin{aligned} \dot{\bar{x}} &= -\bar{x} + \mu\bar{y} \\ \dot{\bar{y}} &= \frac{1}{\gamma}\bar{y}(\gamma - \bar{y})\left(\frac{1}{2} - \bar{x} + (\mu - \frac{1}{\gamma})\bar{y}\right) \end{aligned}$$

In general, is it true that the tree level mean is the equation corresponding to all first order response terms, and the tree propagator satisfies the linearization of the tree mean? Yes. Essentially. The part about the mean is true. Say $\dot{X} = F(X)$ is the tree level mean equation. Then the tree level propagator satisfies $\dot{\Delta}(t, t') - dF(X)\Delta(t, t') = \delta(t - t')$.

Appendix B

Approximating $\exp(t\Gamma)$

The purpose of this appendix is to justify the approximation of $\exp(t\Gamma)$ in (8.34). Suppose $W \approx uv^T$ is a rank 1 approximation to W. I am assuming that the system is stationary so $\bar{g}$ is constant. Let $D_{\phi'}$ be the diagonal matrix where the i^{th} diagonal entry is $\phi'(\bar{g}^i + b^i)$, and define

$$\beta_{chn} = v^T D_{\phi'} u. \tag{B.1}$$

Recall that Γ is defined by (8.29)

$$\Gamma = \frac{1}{\tau}\left(I - \kappa D_{\phi'} W\right), \tag{B.2}$$

where I is the identity matrix. The claim is that

$$\exp(t\Gamma) \approx \exp\left(\frac{t}{\tau}\right)\left(I + \left(\exp\left(-\frac{\kappa\beta_{chn} t}{\tau}\right) - 1\right)\frac{1}{\beta_{chn}} D_{\phi'} W\right). \tag{B.3}$$

The key to this expression is the approximation

$$\Gamma^n \approx \frac{1}{\tau^n}\left(I + \frac{(1-\kappa\beta_{chn})^n - 1}{\beta_{chn}} D_{\phi'} W\right), \tag{B.4}$$

the main ingredient of which is

$$(D_{\phi'} W)^2 \approx D_{\phi'} u v^T D_{\phi'} u v^T \approx \beta_{chn} D_{\phi'} W. \tag{B.5}$$

To justify (B.4) I'll proceed by induction. It is clearly true for the case $n = 0$. Suppose it holds for some n. Then applying (B.5) we have

$$\begin{aligned}
\Gamma^{n+1} &= \Gamma^n \Gamma \\
&\approx \frac{1}{\tau^n}\left(I + \frac{((1-\kappa\beta_{chn})^n - 1)}{\beta_{chn}} D_{\phi'}W\right)\frac{1}{\tau}\left(I - \kappa D_{\phi'}W\right) \\
&= \frac{1}{\tau^{n+1}}\left(I + \left(\frac{(1-\kappa\beta_{chn})^n - 1}{\beta_{chn}} - \kappa\right) D_{\phi'}W - \kappa\left(\frac{(1-\kappa\beta_{chn})^n - 1}{\beta_{chn}}\right)(D_{\phi'}W)^2\right) \\
&\approx \frac{1}{\tau^{n+1}}\left(I + \frac{1}{\beta_{chn}}\left((1-\kappa\beta_{chn})^n - 1 - \kappa\beta_{chn} - \kappa\beta_{chn}((1-\kappa\beta_{chn})^n - 1)\right) D_{\phi'}W\right) \\
&= \frac{1}{\tau^{n+1}}\left(I + \frac{((1-\kappa\beta_{chn})^{n+1} - 1)}{\beta_{chn}} D_{\phi'}W\right).
\end{aligned}$$

With this in hand we easily compute

$$\exp(t\Gamma) \approx \exp\left(\frac{t}{\tau}\right)\left(I + \left(\exp\left(-\frac{\kappa\beta_{chn}t}{\tau}\right) - 1\right)\frac{1}{\beta_{chn}} D_{\phi'}W\right).$$