Reconstructing and Controlling Nonlinear Complex Systems

by

Ri-Qi Su

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved October 2015 by the
Graduate Supervisory Committee:

Ying-Cheng Lai, Co-Chair
Xiao Wang, Co-Chair
Daniel Bliss
Cihan Tepedelenlioglu

ARIZONA STATE UNIVERSITY

December 2015

ABSTRACT

The power of science lies in its ability to infer and predict the existence of objects from which no direct information can be obtained experimentally or observationally. A well known example is to ascertain the existence of black holes of various masses in different parts of the universe from indirect evidence, such as X-ray emissions. In the field of complex networks, the problem of detecting hidden nodes can be stated, as follows. Consider a network whose topology is completely unknown but whose nodes consist of two types: one accessible and another inaccessible from the outside world. The accessible nodes can be observed or monitored, and it is assumed that time series are available from each node in this group. The inaccessible nodes are shielded from the outside and they are essentially "hidden." The question is, based solely on the available time series from the accessible nodes, can the existence and locations of the hidden nodes be inferred? A completely data-driven, compressive-sensing based method is developed to address this issue by utilizing complex weighted networks of nonlinear oscillators, evolutionary game and geospatial networks.

Both microbes and multicellular organisms actively regulate their cell fate determination to cope with changing environments or to ensure proper development. Here, the synthetic biology approaches are used to engineer bistable gene networks to demonstrate that stochastic and permanent cell fate determination can be achieved through initializing gene regulatory networks (GRNs) at the boundary between dynamic attractors. This is experimentally realized by linking a synthetic GRN to a natural output of galactose metabolism regulation in yeast. Combining mathematical modeling and flow cytometry, the engineered systems are shown to be bistable and that inherent gene expression stochasticity does not induce spontaneous state transitioning at steady state. By interfacing rationally designed synthetic GRNs with background gene regulation mechanisms, this work investigates intricate properties

of networks that illuminate possible regulatory mechanisms for cell differentiation and development that can be initiated from points of instability.

To my wife Jie and my parents.

ACKNOWLEDGMENTS

I would like to express my greatest gratitude to both of my advisors, Dr. Ying-Cheng Lai and Dr. Xiao Wang, for their continuous support and mentoring over these few years. Both advisors have educated and helped me throughout my PhD studies by their great passions on research and personality on cooperation.

I also want to take this opportunity to thank Dr. Wenxu Wang, Dr. Liang Huang and Dr. Zigang Huang the former postdocs in Department of Electrical Engineering, and the former postdoc Dr. Min Wu in Department of Biomedical Engineering, for their insightful discussions and warmest encourage during my study.

I will always appreciate the free and productive research environment built by our Chaos group members, Dr. Rui Yang, Dr. Xuan Ni, Yuzhong Chen, Lei Ying, Guanglei Wang, Lezhi Wang, Hongya Xu and Junjie Jiang. It made me feel so supportive and encouraged to discuss and collaborate with them all these years, and it will make our friendship last forever.

I want to thank Dr. Celso Grebogi for providing helpful suggestions and constructive critiques on many of my projects.

My special thanks to the committee members, Dr. Daniel Bliss and Dr. Cihan Tepedelenlioglu, for their precious time as well as invaluable comments and advices, which helped me improve this dissertation.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

INTRODUCTION

1.1 Reconstruct and Control Nonlinear Networks

Complex network is actually a random graph, which however emphasize the rich characteristics in its topology and nodal dynamics. Many of the complex systems can be abstracted as complex networks, such social network[1], citation relationship between scientists[2], metabolic process [3] and gene regularity [4] in biology systems. These networks have varying degree distributions[5], or number of neighborhoods, comparing to traditional random graphs. In certain networks, there exist some 'hub' nodes having extremely large degree than the other nodes. Their dynamics are usually nonlinear, which can exhibit synchronization[6], cascading[7] and other collective behaviors. There are many exist works focusing on how the network topologies will affect their dynamical behaviors[8, 9].

Reverse engineering of complex networks to uncover network topologies from experimental time series of their dynamical behaviors, is a problem of tremendous interest with significant applications [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21]. Earlier examples include reconstruction of gene regulation networks [11] from gene expression data and identification of neuronal interactions based on spike classification methods [12, 13, 14]. More recently, a number of methods for network reconstruction have been proposed, which include reverse engineering of coupled differential equations [15], response-dynamics-based method for coupled phase oscillators [16], phase-space reconstruction based on optimization [17], noise-induced scaling law [18], noise-induced dynamical correlation [19], random phase resetting [20] and inner com-

position alignment [21]. While these methods can successfully determine the network structure, they are unable to determine two pieces of key information needed for predicting the emergence of synchronization: the interaction strength among nodes and the nodal dynamical equations.

To address this challenge, compressive-sensing (CS) [22, 23, 24, 25, 26] based method was proposed and it can uncover not only the full topology of the underlying network, but also the detailed nodal dynamics and link weights (interaction strengths)[27, 28, 28, 29, 30].It has many unique features, such as: (1) it is completely data driven; (2) it can give an accurate estimate of all system parameters; (3) it can lead to faithful reconstruction of the full network structure, even for large networks; and (4) it requires a minimal data amount.

1.2 Compressive Sensing

1.2.1 Sparse Regression

The problem of CS can be stated as follows. Given a low-dimensional measurement vector $\mathbf{X} \in \mathbf{R}^M$, one seeks to reconstruct the much higher-dimensional vector $\mathbf{a} \in \mathbf{R}^N$ according to:

$$\mathbf{X} = \mathbf{G} \cdot \mathbf{a}, \tag{1.1}$$

where $N \gg M$ and $\mathbf{G}$ is an $M \times N$ projection matrix. A sufficiently sparse vector $\mathbf{a}$ can be reconstructed by solving the following convex optimization problem [22, 23, 24, 25, 26]:

$$\min \|\mathbf{a}\|_1, \text{ s.t.} \mathbf{X} = \mathbf{G} \cdot \mathbf{a}, \tag{1.2}$$

where the $\|\mathbf{a}\|_1 = \sum_{i=1}^{N} |\mathbf{a}_i|$ is the l_1 norm of vector $\mathbf{a}$.

Here sufficiently sparse is defined by the Restricted isometry property (RIP), which address the approximated orthonormality of a random matrix on sparse vectors[22].

The given matrix $\mathbf{G}$ is said to statisfy RIP when there exist a constant δ, that:

$$(1-\delta)\|a\|_2^2 \leq \|Ga\|_2^2 \leq (1+\delta)\|a\|_2^2, \tag{1.3}$$

where vector y is an arbitrary given sparce vector with P non-zero terms.

1.2.2 Inaccurate Measurement

When the given measurement is not accurate, says, linear equation $\mathbf{X} = \mathbf{G} \cdot \mathbf{a} + \xi$, and ξ is a Q-dimension random variable and satisfies Gaussian distribution of zero mean and variance as σ, the stable recovery of the P-dimension sparse vector $\mathbf{a}$ is achievable, according to [22]. If the unknown vector $\mathbf{a}$ is sufficiently sparse, we can reconstruct it by solving the following l_1 regularization problem:

$$\min \|\mathbf{a}\|_1, \text{ subject to } \|\mathbf{G} \cdot \mathbf{a} - \mathbf{X}\|_2 \leq \epsilon, \tag{1.4}$$

where the l_2 norm is $\|\mathbf{X}\|_2 = \sum_{i=1}^{n} |x_i^2|$. ϵ is the size of the error term ξ. The reconstructed vector $\bar{\mathbf{a}}$ is proved to be within the noise level as $\|\bar{\mathbf{a}} - \mathbf{a}\| \leq C \cdot \epsilon$, and C is a constant.

1.2.3 Network Reconstruction Using Compressive Sensing

Now I will explain the general procedure to reconstruct network dynamics and topologies from time series using compressive sensing[27, 28, 28, 29, 30]. Consider a general dynamics for an isolated oscillator in the network, its dynamic can be written as:

$$\dot{\mathbf{x}}_i(t) = \mathbf{F}_i[\mathbf{x}_i(t)] + \mathbf{S}_i(t), \tag{1.5}$$

where $\mathbf{x}_i \in R^m$ is an m-dimensional dynamical variable and $\mathbf{S}_i(t)$ denotes the external driving. Then the complete network equations can then be written as:

$$\dot{\mathbf{x}}_i = \mathbf{F}_i(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij} \cdot [\mathbf{H}(\mathbf{x}_j) - \mathbf{H}(\mathbf{x}_i)] + \mathbf{S}_i(t), \tag{1.6}$$

where $\mathbf{W}_{ij} \in R^{m\times m}$ is the weighted coupling matrix between node i and node j and $\mathbf{H}$ is the coupling function.

Our goal is to reconstruct the nodal velocity field $\mathbf{F}_i$ and all of the coupling matrices $\mathbf{W}$ using time series $\mathbf{x}(t)$ and the given driving signal $\mathbf{S}(t)$. First, we group all terms directly associated with node i into $\mathbf{F}'_i(\mathbf{x}_i)$, by defining:

$$\mathbf{F}'_i(\mathbf{x}_i) \equiv \mathbf{F}_i(\mathbf{x}_i) - \mathbf{H}(\mathbf{x}_i) \cdot \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij}.$$

We have:

$$\dot{\mathbf{x}}_i = \mathbf{F}'_i(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij}\mathbf{H}(\mathbf{x}_j) + \mathbf{S}_i(t). \tag{1.7}$$

Then, we choose a suitable base and expand $\mathbf{F}'(\mathbf{x}_i)$ into the following form:

$$\mathbf{F}'_i(\mathbf{x}_i) = \sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i), \tag{1.8}$$

where $\tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i)$ are a set of orthogonal and complete base functions, which are chosen such that the coefficients $\tilde{\mathbf{a}}_i^{(\gamma)}$ are sparse. While the coupling function $\mathbf{H}(\mathbf{x}_i)$, if it is nonlinear, can be expanded in a similar manner, for notational convenience, we assume that they are linear: $\mathbf{H}(\mathbf{x}_i) = \mathbf{x}_i$. We then have:

$$\dot{\mathbf{x}}_i = \sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij} \cdot \mathbf{x}_j + \mathbf{S}_i(t), \tag{1.9}$$

where all of the coefficients $\tilde{\mathbf{a}}_i^{(\gamma)}$ and $\mathbf{W}_{ij}$ are to be determined from time series $\mathbf{x}_i$ via CS. Specifically, the coefficient vector $\tilde{\mathbf{a}}_i^{(\gamma)}$ determines the nodal dynamics, and the weighted matrices $\mathbf{W}_{ij}$ give the full topology and coupling strengths of the entire network.

Suppose we have measurements of all state variables $\mathbf{x}_i(t)$ at M different values of t and assume further that for each t value, the values of the state variables at a slightly later time, $t + \delta t$, are also available, where $\delta t \ll \Delta t$, so that the derivative vector $\dot{\mathbf{x}}_i$

can be estimated at each time instant. Equation (1.9) for all M time instants can then be written in the following matrix form:

$$\mathbf{G}_i = \begin{pmatrix} \tilde{\mathbf{g}}_i(t_1) & \mathbf{x_1}(t_1) & \cdots & \mathbf{x}_k(t_1) & \cdots & \mathbf{x}_N(t_1) \\ \tilde{\mathbf{g}}_i(t_2) & \mathbf{x_1}(t_2) & \cdots & \mathbf{x}_k(t_2) & \cdots & \mathbf{x}_N(t_2) \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ \tilde{\mathbf{g}}_i(t_M) & \mathbf{x_1}(t_M) & \cdots & \mathbf{x}_k(t_M) & \cdots & \mathbf{x}_N(t_M) \end{pmatrix}, \tag{1.10}$$

where the index k in $\mathbf{x}_k(t)$ runs from one to N, $k \neq i$, and each row of the matrix is determined by the available time series at one instant of time. The derivatives at different times can be written in a vector form as: $\mathbf{X}_i = [\dot{\mathbf{x}}_i(t_1), \cdots, \dot{\mathbf{x}}_i(t_M)]^T$. The coefficients from the functional expansion and the weights associated with all links in the network, which are to be determined, can be combined concisely into a vector $\mathbf{a}_i$, as follows:

$$\mathbf{a}_i = [\tilde{\mathbf{a}}_i, \mathbf{W}_{1i}, \cdots, \mathbf{W}_{i-1,i}, \mathbf{W}_{i+1,i}, \cdots, \mathbf{W}_{N,i}]^T. \tag{1.11}$$

where $[\cdot]^T$ denotes the transpose. For a properly chosen expansion base and a general complex network whose connections are typically sparse, the vector $\mathbf{a}_i$ to be determined is sparse, as well. Finally, Equation (1.9) can be written in the standard CS form as:

$$\mathbf{X}_i = \mathbf{G}_i \cdot \mathbf{a}_i + \mathbf{S}_i(t), \tag{1.12}$$

a linear equation in which the dimension of the unknown coefficient vector $\mathbf{a}_i$ can be much larger than that of $\mathbf{X}_i$, and the measurement matrix $\mathbf{G}_i$ will have many more columns than rows. In a conventional sense, this equation is ill defined, but since $\mathbf{a}_i$ is sparse, insofar as its number of non-zero coefficients is smaller than the dimension of $\mathbf{X}_i$, the vector $\mathbf{a}_i$ can be uniquely and efficiently determined by CS [22, 31, 25, 26, 24].

1.3 Nonlinear Gene Networks

Bistability and the binary decision-making it imparts have been widely observed and hypothesized as one of the possible mechanisms for cell fate determination [32, 33, 34]. Previous studies of bistable systems have attributed this binary decision-making to either (i) random and reversible state transitioning - cells spontaneously and randomly switching back and forth between two states without environmental perturbations [35, 36, 37, 38], or (ii) deterministic and irreversible state transitioning - cells uniformly and irreversibly choosing one of two states in response to external signals [39, 40, 41]. Few mechanisms, however, have been proposed to explain the scenario of random and yet irreversible cell fate determination, commonly seen in development and cell differentiation. Recent studies [42, 43, 44] show examples of stochastic and irreversible cell differentiation in multicellular organisms, and have identified that the central regulatory motif driving these stochastic differentiations is a mutual inhibitory GRN, a common topological module that can generate bistability. However, with neither fluctuating environmental cues nor spontaneous state transitioning identified in these cases, an understanding of how cells differentiate stochastically and irreversibly into distinct subpopulations remains elusive, especially when under the tight control of GRNs.

Synthetic gene networks provide an effective platform to probe the otherwise intractable properties of common network motifs and uncover novel mechanisms for counterintuitive observations [4]. Such investigations are impossible in their natural settings where the complex interconnectivity of native GRNs acts as a major barrier to detailed analysis. Synthetic gene networks, on the other hand, are rationally designed and constructed to realize core topological modules of GRN in vivo without interference from auxiliary connections. They can therefore be studied in isolation

at great detail to reveal novel insights into the design and working of biological systems and processes [45], such as gene expression noise [35, 46, 47, 48, 49, 50, 51], multistability [39, 41, 52], oscillations [53, 54, 55, 56], intracellular signaling [57, 58], intercellular communications [59, 60], and multicellular pattern formation [61, 62].

Chapter 2

IDENTIFY CHAOTIC OSCILLATORS FROM NEURAL NETWORKS

In this project, we address the problem of the data-based identification of a subset of chaotic elements embedded in a network of nonlinear oscillators. In particular, given such a network, we assume that time series can be measured from each oscillator. The oscillators, when isolated, are not identical in that their parameters are different, so dynamically, they can be in distinct regimes. For example, all oscillators can be described by differential equations of the same mathematical form, but with different parameters. Consider the situation where only a small subset of the oscillators are chaotic and the remaining oscillators are in dynamical regimes of regular oscillations. Due to mutual couplings among the oscillators, the measured time series from most oscillators would appear random. The challenge is to identify the small subset of originally ("truly") chaotic oscillators.

2.1 Background and Motivation

The problem of identifying chaotic elements from a network of coupled oscillators arises in biological systems and biomedical applications. For example, consider a network of coupled neurons that exhibit regular oscillations in a normal state. In such a state, the parameters of each isolated neuron are in the regular regime. Under external perturbations or slow environmental influences, the parameters of some neurons can drift into the chaotic regime. When this occurs, the whole network would appear to behave chaotically, which may correspond to a certain disease. The virtue of nonlinearity stipulates that the irregular oscillations at the network level can emerge even if only a few oscillators have gone "bad". It is thus desirable to be able to pin

down the origin of the ill-behaved oscillators—the few chaotic neurons among a large number of healthy ones.

One might attempt to use the traditional approach of time-delayed coordinate embedding to reconstruct the phase space of the underlying dynamical system [63, 64, 65] and then to compute the Lyapunov exponents [66, 67]. However, since we are dealing with a network of nonlinear oscillators, the phase-space dimension is high and an estimate of the largest Lyapunov exponent would only indicate if the whole coupled system is chaotic or nonchaotic, depending on the sign of the estimated exponent. In principle, using time series from any specific oscillator(s) would give qualitatively the same result. Thus, the traditional approach cannot give an answer as to which oscillators are chaotic when isolated.

There were previous efforts in nonlinear systems identification and parameter estimation for coupled oscillators and spatiotemporal systems, such as the auto-synchronization method [68]. There were also works on revealing the connection patterns of networks. For example, a methodology was proposed to estimate the network topology controlled by feedback or delayed feedback [69, 70, 71]. Network connectivity can be reconstructed from the collective dynamical trajectories using response dynamics, as well [72, 73]. In addition, the approach of random phase resetting was introduced to reconstruct the details of the network structure [74]. For neuronal systems, there was a statistical method to track the structural changes [75, 76]. While many of these previous methods require complete or partial information about the dynamical equations of the isolated nodes and their coupling functions, completely data-driven and model-free methods exist. For example, the network structure can be obtained by calculating the causal influences among the time series based on, for example, the Granger causality method [77, 78], the transfer-entropy method [79] or the method of inner composition alignment [80]. However, such causality-based meth-

ods are unable to reveal information about the dynamical equations of the isolated nodes. There were regression-based methods [81] for systems identification based on, for example, the least-squares approximation through the Kronecker-product representation [82], which would require large amounts of data. (Due to the L_1 nature of compressive sensing [23, 31, 25, 26, 24], the data requirement in our method can be significantly relaxed.) The unique features of our method are: (1) it is completely data driven; (2) it can give an accurate estimate of all system parameters; (3) it can lead to faithful reconstruction of the full network structure, even for large networks; and (4) it requires a minimal data amount. While some of these features are shared by previous methods, no single previous method possesses all of these features.

Here, we develop a method to address the problem of identifying a subset of ill-behaved chaotic elements from a network of nonlinear oscillators, the majority of them being regular. The basic mathematical framework underlying our method is compressive sensing (CS), a paradigm for high-fidelity signal reconstruction using only sparse data [23, 31, 25, 26, 24]. The CS paradigm was originally developed to solve the problem of transmitting extremely large data sets, such as those collected from large-scale sensor arrays. Because of the extremely high dimensionality, direct transmission of such data sets would require a very broad bandwidth. However, there are many applications in which the data sets are sparse. To be concrete, say a data set of N points is represented by an $N \times 1$ vector, $\mathbf{a}$, where N is a very large integer. Then, $\mathbf{a}$ being sparse means that most of its entries are zero and only a small number of k entries are non-zero, where $k \ll N$. One can use a random matrix $\mathbf{G}$ of dimension $M \times N$ to obtain an $M \times 1$ vector $\mathbf{X}$: $\mathbf{X} = \mathbf{G} \cdot \mathbf{a}$, where $M \sim k$. Because the dimension of $\mathbf{X}$ is much lower than that of the original vector $\mathbf{a}$, transmitting $\mathbf{X}$ would require a much smaller bandwidth, provided that $\mathbf{a}$ can be reconstructed at the other end of the communication channel. Under the constraint that the vector to be reconstructed is

sparse, the feasibility of faithful reconstruction is guaranteed mathematically by the CS paradigm [23, 31, 25, 26, 24]. In the past decade, CS has been exploited in a large variety of applications, ranging from optical image processing [83] and reconstruction of nonlinear dynamical and complex systems [29, 30] to quantum measurements [84].

It has been shown in a series of recent papers [29, 30, 28, 85, 86, 87] that the detailed equations and parameters of nonlinear dynamical systems and complex networks can be accurately reconstructed from short time series using the CS paradigm. Here, we extend this approach to a network of coupled, mixed nonchaotic and chaotic neurons. We demonstrate that, by formulating the reconstruction task as a CS problem, the system equations and coupling functions, as well as all of the parameters can be obtained accurately from sparse time series. Using the reconstructed system equations and parameters for each and every neuron in the network and setting all of the coupling parameters to zero, a routine calculation of the largest Lyapunov exponent can unequivocally distinguish the chaotic neurons from the nonchaotic ones.

We remark on the generality of our compressive sensing-based method. Insofar as time series from all dynamical variables of the system are available and a suitable mathematical base can be found in which the nodal and coupling functions can be expanded in terms of a sparse number of terms, the whole system, including all individual nodal dynamics, can be accurately reconstructed. With the reconstructed individual nodal equations, chaotic neurons can be identified through routine calculation of the largest Lyapunov exponent.

2.2 Data Driven Method to Identify and Control Chaotic Oscillactors

Figure 2.1(a) shows schematically a representative coupled neuronal network. Consider a pair of neurons, one chaotic and another nonchaotic when isolated (say 1 and 10, respectively). When they are placed in a network, due to coupling, the time

series collected from both will appear random and qualitatively similar, as shown in Figure 2.1b,c. It is visually quite difficult to distinguish the time series and to ascertain which node is originally chaotic and which is regular. The difficulty is compounded by the fact that the detailed coupling scheme is not known *a priori*. Say that the chaotic behavior leads to the undesirable function of the network and is to be suppressed. A viable and efficient method is to apply small pinning controls [88, 89, 90, 91] to the relatively few chaotic neurons to drive them into some regular regime. (Here, we assume the network is such that, when all neurons are regular, the collective dynamics is regular. That is, we exclude the uncommon, but not unlikely, situation that a network system of coupled regular oscillators would exhibit chaotic behaviors.) Accurate identification of the chaotic neurons is thus key to implementing the pinning control strategy.

Given a neuronal network, our aim is thus to locate all neurons that are originally chaotic and neurons that are potentially likely to enter into a chaotic regime when they are isolated from the other neurons or when the couplings among the neurons are weakened. Our approach consists of two steps. Firstly, we employ the CS framework to estimate, from measured time series only, the parameters in the FHN equation for each neuron, as well as the network topology and various coupling functions and weights. As will be shown below, this can be done by expanding the nodal dynamical equations and the coupling functions into some suitable mathematical base, as determined by the specific knowledge of the actual neuronal dynamical system, and then casting the problem into that of determining the sparse coefficients associated with various terms in the expansion. The nonlinear system identification problem can then be solved using some standard CS algorithm. Secondly, we set all coupling parameters to zero and analyze the dynamical behaviors of each and every individual

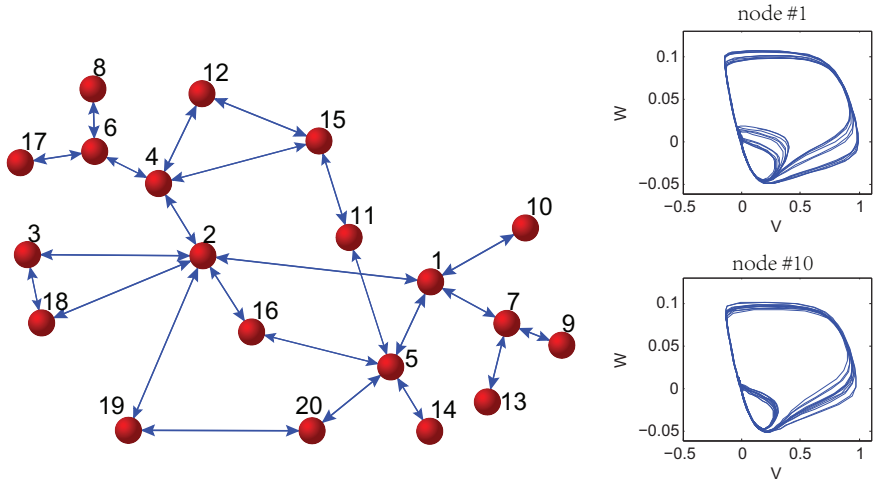

Figure 2.1: (a) Schematic illustration of a small neuronal network, where the dynamics of each neuron is mathematically described by the FitzHugh–Nagumo (FHN) equations. (b,c) Dynamical trajectories of two neurons from the coupled system, one being chaotic when isolated and another regular, respectively. The trajectories give little hint as to which one is originally chaotic and which one is regular, due to the coupling. Specifically, Neuron 1 is originally chaotic (by setting parameter $a = 0.42$ in the FHN equation), while all other neurons are regular (their values of the corresponding parameter in the FHN equation are chosen uniformly from the interval $[0.43, 0.45]$).

neuron by calculating the Lyapunov exponents. Those with positive largest exponent are identified as chaotic.

A typical time series from a neuronal network consists of a sequence of spikes in the time evolution of the cell membrane potential. We demonstrate that our CS-based reconstruction method works well even for such spiky time series. We also analyze the dependence of the reconstruction accuracy on the data amount and show that only limited data are required to achieve high accuracy in reconstruction.

2.3 Simulation Results on FHN Networks

The FHN model, a simplified version of the biophysically detailed Hodgkin–Huxley model [92], is a mathematical paradigm for gaining significant insights into a variety of dynamical behaviors in real neuronal systems [93, 94]. For a single, isolated neuron,

the corresponding dynamical system is described by the following two-dimensional, nonlinear ordinary differential equations:

$$\begin{aligned} \frac{dV}{dt} &= \frac{1}{\delta}[V(V-a)(1-V)-W], \\ \frac{dW}{dt} &= V - W - b + S(t), \end{aligned} \tag{2.1}$$

where V is the membrane potential, W is the recover variable, $S(t)$ is the driving signal (e.g., periodic signal) and a, b and δ are parameters. The parameter δ is chosen to be infinitesimal, so that $V(t)$ and $W(t)$ are "fast" and "slow" variables, respectively. Because of the explicitly time-dependent driving signal $S(t)$, Equation (2.1) is effectively a three-dimensional dynamical system, in which chaos can arise [95]. For a network of FHN neurons, the equations are:

$$\begin{aligned} \frac{dV_i}{dt} &= \frac{1}{\delta}[V_i(V_i-a)(1-V_i)-W_i] + \sum_{i=1}^{N} c_{ij}(V_j - V_i) \\ \frac{dW_i}{dt} &= V_i - W_i - b + S(t), \end{aligned} \tag{2.2}$$

where c_{ij} is the coupling strength (weight) between the i-th and the j-th neurons (nodes). For $c_{ij} = c_{ji}$, the interactions between any pair of neurons are symmetric, leading to a symmetric adjacency matrix for the network. For $c_{ij} \neq c_{ji}$, the network is asymmetrically weighted.

We consider the FHN model with sinusoidal driving: $S(t) = r \sin \omega_0 t$. The model parameters are $r = 0.32$, $\omega_0 = 15.0$, $\delta = 0.005$ and $b = 0.15$. For $a = 0.42$, an individual neuron exhibits chaos. The time series are generated by the fourth-order Runge–Kutta method with step size $h = 10^{-4}$. We sample three consecutive measurements at time interval $\tau = 0.05$ apart and then use a standard two-point formula to calculate the derivative. Representative chaotic time series and the corresponding dynamical trajectory are shown in Figure 2.2.

We first present the reconstruction result for an isolated neuron, by setting to zero all coupling terms in Eq. 2.2. Following the method introduced in Sec. I, Eq. 2.2 can be written in the standard CS form as:

$$\mathbf{X}_i = \mathbf{G}_i \cdot \mathbf{a}_i + \mathbf{S}_i(t), \tag{2.3}$$

Where the vector $\mathbf{a}_i$ to be determined then contains the unknown parameters associated with a single neuron only. We choose power series of order four as the expansion base, so that there are 17 unknown coefficients to be determined. We use 12 data points generated from a random starting point. The results of the reconstruction are shown in Figure 2.3a,b for variables V and W, respectively. The last two coefficients associated with each variable represent the strength of the driving signal. Since only the variable W receives sinusoidal input, the last coefficient in W is nonzero. By comparing the positions of nonzero terms and our previously assumed vector form, $\mathbf{g}_i(t)$, we can fully reconstruct the dynamical equations of any isolated neuron. In particular, we see from Figure 2.3a,b that all estimated coefficients agree with their respective true values. Figure 2.3c shows how the estimated coefficients converge to the true values as the number of data points is increased. We see that, for over 10 data points, we can already reconstruct faithfully all of the parameters.

Next, we consider the network of coupled FHN neurons as schematically shown in Figure 2.1a, where the coupling weights among various pairs of nodes are uniformly distributed in the interval $[0.3, 0.4]$. The network is random with connection probability $p = 0.04$. From time series, we construct the CS matrix for each variable of all nodes. Since the couplings occur among the variables V of different neurons, the strengths of all incoming links can be found in the unknown coefficients associated with different V variables. Extracting all coupling terms from the estimated coefficients, we obtain all off-diagonal terms in the weighted adjacency matrix.

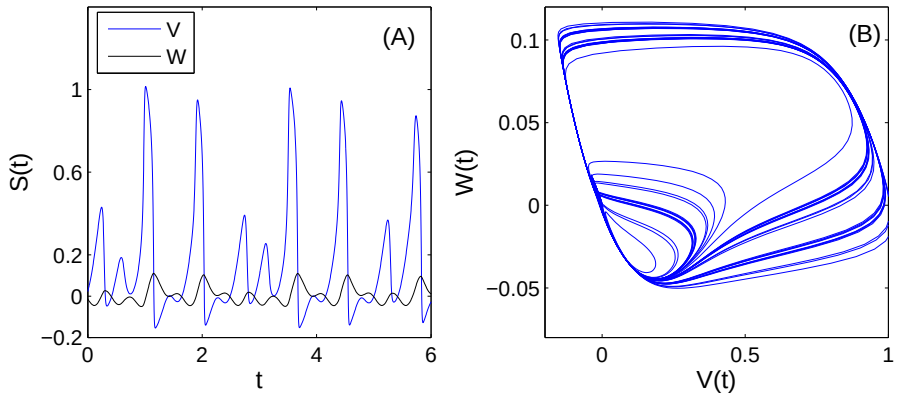

Figure 2.2: (a) Chaotic time series of the membrane potential V and recovery variable W from a single neuron for $a = 0.42$; and (b) the corresponding dynamical trajectory.

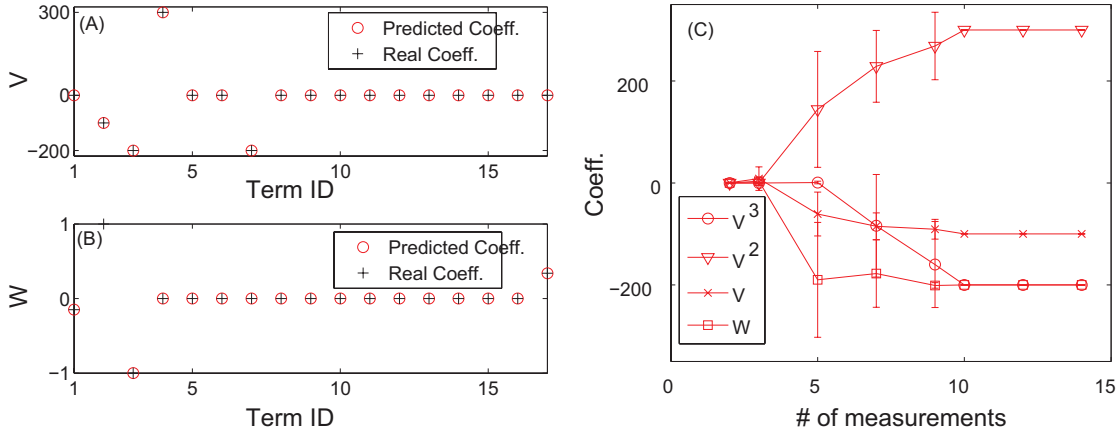

Figure 2.3: (a,b) Predicted coefficients from compressive sensing (CS) and a comparison with the actual parameter values in the dynamical equations of variables V and W. The number of data points used is 12. (c) Predicted parameters for a single neuron as the number of data points is increased. The sampling interval is $\Delta t = 0.05$. All results are averaged over 10 independent time series.

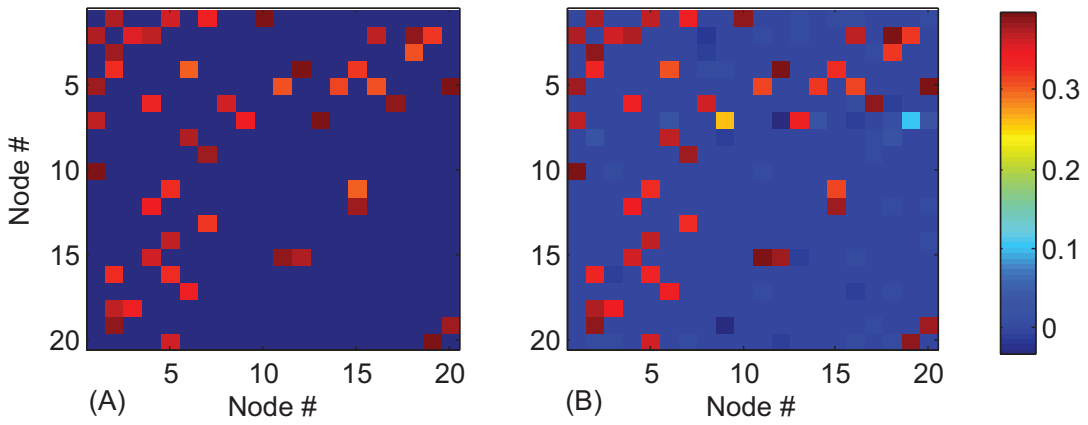

Figure 2.4: For the network in Figure 2.1a, (a) the actual and (b) estimated weighted adjacency matrix. The normalized data amount used in the reconstruction is $R_m = 0.7$.

To assess the reconstruction accuracy, we define E_{nz} as the average normalized difference between the non-zero terms in the estimated coefficients and the real values:

$$E_{nz} = \frac{1}{M_{nz}} \sum_{k=1}^{M_{nz}} \| \frac{c'_k - c_k}{c_k} \|,$$

where M_{nz} is the number of non-zero terms in the actual coefficients, c'_k and c_k are the k-th nonzero terms in the estimated coefficients and the true one, respectively. For convenience, we define R_m as the relative number of data points normalized by the number of total unknown coefficients. Figure 2.4 shows the reconstructed adjacency matrix as compared with the real one for $R_m = 0.7$. We see that our method can predict all links correctly, in spite of the small errors in the predicted weight values. The errors are mainly due to the fact that there are large coefficients in the system equations, but the coupling weights are small.

Using the weighted adjacency matrix, we can identify the coupling terms in the vector function $\mathbf{F}'_i(\mathbf{x}_i)$, so as to extract the terms associated with isolated nodal velocity field $\mathbf{F}_i$. We can then determine the value of parameter a and calculate the largest Lyapunov exponent for each individual neuron. The results are shown in

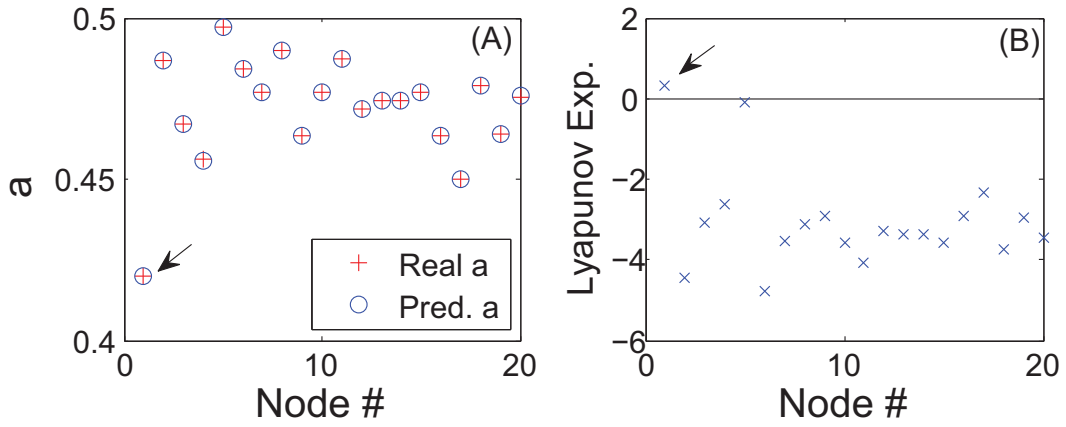

Figure 2.5: (a) Estimated values of parameter a for different neurons (red circles), as compared with the actual values (black crosses). The random network size is $N = 20$ with connection probability $p = 0.04$. The normalized data amount used in reconstruction is $R_m = 0.7$. (b) The largest Lyapunov exponents calculated from the reconstructed system equations. The reference line denotes a null value.

Figure 2.5a,b. We see that, for this example, Neuron 1 has a positive largest exponent, while the largest exponents for all others are negative, so 1 is identified as the only chaotic neuron among all neurons in the network.

Next, we discuss the relationship between reconstruction error and data requirement. As shown in Figure 2.6, for different network sizes N, the reconstruction error decreases with R_m. For R_m larger than a threshold, the normalized error E_{nz} is small. For $N = 40$, the threshold is about 0.6, and it is 0.5 for $N = 60$. That is because, for fixed connection probability, a larger network will have more sparse connections, requiring a smaller value of R_m for accurate reconstruction.

Finally, we study the performance of our method with respect to systematically varying of network size and edge density. As with any method, larger networks require more computation. We study networks of a size up to $N = 100$ nodes. Figure 2.7a shows the normalized error associated with the nonzero terms, E_{nz}, for different net-

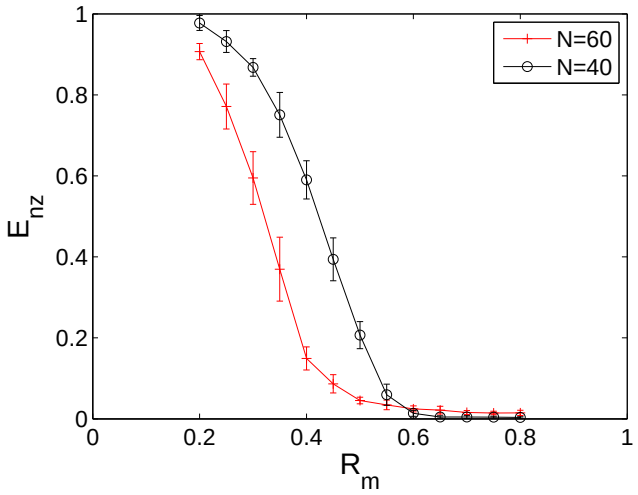

Figure 2.6: For neural networks of size $N = 40$ and $N = 60$, the normalized error associated with non-zero terms, E_{nz}, as a function of normalized data amount R_m. The results are averaged over 10 independent measurement realizations.

work sizes N and normalized data amount R_m. For a given network size, similar to Figure 2.6, E_{nz} gradually decreases to a certain low level as the relative data amount R_m is increased. We further observe that smaller values of R_m are required to reconstruct larger networks of the same connecting probability P. Note that R_m is the relative data amount defined with respect to the number of unknown coefficients, so for larger networks, the absolute data amount required actually increases. In Figure 2.7b, we show the contour plot of values of E_{nz} in the parameter plane (R_m, P) for a fixed network size ($N = 60$). We see that, for a fixed value of R_m, as P is increased, the error E_{nz} also increases, which is anticipated, as denser networks lead to a denser projection matrix in compressive sensing.

2.4 Conclusions

We develop a completely data-driven method to detect chaotic elements embedded in a network of nonlinear oscillators, where such elements are assumed to be relatively

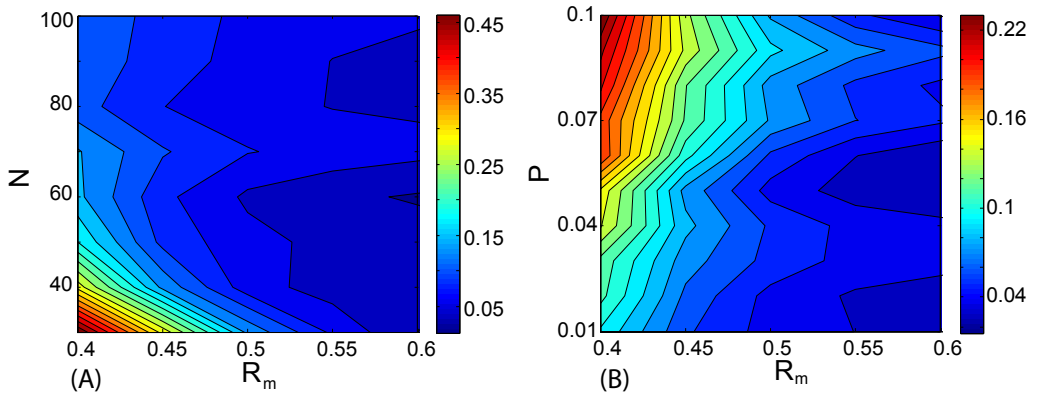

Figure 2.7: (a) For a random network of fixed connecting probability $p = 0.04$, a contour plot of the normalized error associated with nonzero terms, E_{nz}, in the parameter plane (R_m, N). (b) For a random network of fixed size $N = 60$, a contour plot of E_{nz} in the parameter plane (R_m, P). All results are obtained from 10 independent network realizations. See the text for explanations.

few. From a biomedical perspective, the chaotic elements can be the source of certain diseases, and their accurate identification is desirable. In spite of being only a few, the chaotic oscillators can cause the time series from other, originally regular oscillators to appear random, due to the network interactions among the oscillators. The standard method in nonlinear time series, the method of delay-coordinate embedding, cannot be used to identify the local chaotic elements, because the method can give information only about the global dynamics. For example, one can attempt to estimate the largest Lyapunov exponent by using time series, either from a chaotic oscillator or from an originally regular oscillator, and the embedding method would yield qualitatively or even quantitatively similar results. Our compressive sensing-based method, however, overcomes such difficulties by generating an accurate estimate of all system equations, which include the local dynamical equations of each individual node and all coupling functions. Isolating the coupling functions from the local velocity fields, we can obtain the original dynamical equations for each individual oscillator, enabling efficient calculation of the Lyapunov exponents for all oscillators and, consequently,

accurate identification of the chaotic oscillators. We illustrate this methodology by using model networks of FHN neurons. One key virtue of compressive sensing, namely the low data requirement, enables us to accomplish the task of identifying chaos with short time series. Our method is generally applicable to any nonlinear dynamical networks, insofar as time series from the oscillators are available.

Comparing with our previous works on compressive sensing-based nonlinear system identification and reverse engineering of complex networks [29, 30, 28, 85, 86, 87], the new technical features of the present work are the following. Firstly, we demonstrate that the compressive sensing-based system identification is effective for spiky time series that are typical of neuronal networks. Secondly, local velocity fields and non-uniform weights of node-to-node interactions can be reconstructed accurately for neuronal networks with both fast and slow variables in the presence of external driving. Thirdly, the method works regardless of the ratio between the number of originally chaotic and nonchaotic oscillators. The great flexibility, the extreme low data requirement and high accuracy make our method appealing for various problems arising from nonlinear system identification, especially in biology and biomedicine.

There are a number of limitations to our method. For example, for any accessible node in the network, time series of all dynamical variables are required. If information from one node or some of the nodes in the network is inaccessible, or "hidden" from the outside world, it is not feasible to recover the nodal dynamical system of such nodes and their neighbors [85, 87]. The "hidden dimensions" problem, in which some dynamical variables are not given, is another obstacle to realistic applications. Our compressive sensing-based method also requires reasonable knowledge about the underlying complex system, so that a suitable mathematical base can be identified for expansions of the various nodal and coupling functions. Further efforts are certainly needed.

Chapter 3

PREDICT COLLECTIVE BEHAVIORS OF WEIGHTED OSCILLACTOR NETWORKS

3.1 Introduction

The most amazing feature of a complex dynamical system consisting of a large number of interacting units (or components) is the emergence of collective dynamics. Indeed, it is this feature of "more is different" [96] which makes complex systems extremely interesting and the study of collective dynamics fundamentally important to many natural and technological systems. Given a complex system, if the underlying mathematical rules or equations are completely known, then *in principle* the possible types of collective dynamics in the system can be predicted and studied, and most existing works on complex systems are of this nature. In realistic applications one may encounter the situation where, for a complex system of interest, the local system equations and the interactions among the components are not known *a priori* but only a set of time series are available. Can one still forecast or anticipate whether a certain type of collective dynamics can potentially occur in the system?

Even when the system equations of a complex system are known, it is still extremely challenging to predict, investigate, and explore the emergence and evolution of collective dynamics. In order to address the issue of time-series based prediction of collective dynamics, one must focus on a relatively well known class of such dynamics. We shall then consider synchronization [97, 98, 99]. Specifically, we shall study coupled-oscillator networks[100], a paradigm for probing and understanding the synchronous behavior of interacting units with nonlinear dynamics. When the

system equations are known, a widely used tool to determine whether synchronization can emerge physically is the master-stability function (MSF) articulated by Pecora and Carroll [99, 100]. In the MSF framework, synchronization under various combinations of network structures and oscillator dynamics can be predicted[100, 101]. For example, given the nodal dynamical equations, possible states of synchronization can be determined, which are basically the possible dynamics on the synchronization manifold. The MSF is nothing but the largest Lyapunov exponent characterizing the transverse stability of the synchronous dynamical state. For a typical nonlinear or chaotic oscillator, there may exist an open interval in the space of some generalized coupling parameter [101], where the MSF is negative so that any point in this interval can lead to stable synchronization. When the network structure is given, the set of eigenvalues of the underlying coupling matrix can be determined. For a network of coupled oscillators, the phase-space dimension can be extremely high, so there can be many transverse subspaces. The set of eigenvalues, after suitable normalization, gives the set of effective generalized coupling parameters associated with all the transverse subspaces. Network synchronization can occur only when all these parameters fall into the interval of negative MSF.

In this project, we propose a general approach to forecasting the emergence of synchronization in complex oscillator networks based on a complete set of time series collected from all components of every oscillator. The specific setting of the problem is, as follows. Assume that at the time of interest the oscillator network is in an asynchronous state and time series from each node in the network can be obtained. Assume further that there exists a parameter characterizing the average coupling strength among the nodes. The question we ask is whether it would be possible to predict that synchronization can or cannot occur when the coupling parameter is allowed to change. Our method consists of two steps. Firstly, we reconstruct the full

topology of the network, together with the coupling strengths and the nodal dynamics, based solely on time series. This is accomplished by casting the prediction or reverse-engineering [102, 103, 104] problem into the framework of compressive sensing, a recently developed, powerful convex optimization paradigm [22, 23, 24, 25, 26] for recovering sparse vectors based on very limited amount of data. Here the relevant vector to be reconstructed originated from both nodal dynamics and topology, which is typically sparse due to the sparsity of complex networks. Secondly, from the predicted nodal dynamics and network structure, we perform synchronizability analysis by using the standard MSF approach. We validate our method by using random *weighted* networks [105] of both continuous-time and discrete-time chaotic systems (e.g., the classical Lorenz system [106] and Hénon map [107]). Our computation and analysis indicate that with only small amount of measured data, the synchronization regions in the parameter space as identified by MSF and the network structure can be accurately predicted, rendering possible inference of synchronous dynamics. The critical data requirement and sampling frequency for different network sizes and degree distributions are studied in detail. The issue of the effect of measurement noise on prediction accuracy is also addressed. In addition, the dependence of data requirement and computational time on the network size are studied. Finally, we speculate on one potential application of our prediction method: controlling coupled oscillators to bring the system to synchronization.

In Sec. 3.2, we describe our compressive-sensing based method for reconstructing weighted complex oscillator networks and for estimating the MSF. In Sec. 3.3, a detailed account of representative examples is presented, together with a systematic analysis of the prediction accuracy, data requirement from different perspectives, effects of network size and noise, and computation time. In Sec. 3.4, we discuss how

possible emergence of synchronous dynamics can be anticipated based on data. In Sec. 3.5, a conclusion and discussions are provided.

3.2 Network System Reconstruction and Synchronizability Analysis

Our method is in fact a combination of two problems: compressive-sensing based reverse engineering of complex networked dynamical systems [29, 30] and synchronizability analysis.

3.2.1 Reverse Engineering of Weighted Complex Networked Dynamical Systems

Reverse engineering of complex networks to uncover network topologies from experimental time series is a problem of tremendous interest with significant applications [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21]. Earlier examples include reconstruction of gene regulation networks [11] from gene expression data and identification of neuronal interactions based on spike classification methods [12, 13, 14]. More recently, a number of methods for network reconstruction have been proposed, which include reverse engineering of coupled differential equations [15], response-dynamics-based method for coupled phase oscillators [16], phase-space reconstruction based on optimization [17], noise-induced scaling law [18], noise-induced dynamical correlation [19], random phase resetting [20] and inner composition alignment [21]. While these methods can successfully determine the network structure, they are unable to determine two pieces of key information needed for predicting the emergence of synchronization: the interaction strength among nodes and the nodal dynamical equations. As will be explained, our compressive-sensing [22, 23, 24, 25, 26] based method can uncover not only the full topology of the underlying network, but also the detailed nodal dynamics and link weights (interaction strengths), making it possible to forecast synchronization.

3.2.2 Stability Analysis for Synchronous Dynamics

After the nodal dynamics and the network structure have been uncovered from time series, we can use the MSF framework to assess the emergence of synchronous dynamics and its stability . For the network system, the synchronous state $\mathbf{x}_1 = \mathbf{x}_2 = \cdots = \mathbf{x}_N = \mathbf{s}$, where $d\mathbf{s}/dt = \mathbf{F}(\mathbf{s})$, is an exact solution. The time evolutions of small variations from the synchronous state, $\delta\mathbf{x}_i(t) \equiv \mathbf{x}_i(t) - \mathbf{s}(t)$, are governed by

$$\frac{d\delta\mathbf{x}_i}{dt} = \mathbf{DF}(\mathbf{s}) \cdot \delta\mathbf{x}_i - \xi \sum_{j=1}^{N} G_{ij}\mathbf{DH}(\mathbf{s}) \cdot \delta\mathbf{x}_j, \tag{3.1}$$

where $\mathbf{DF}(\mathbf{s})$ and $\mathbf{DH}(\mathbf{s})$ are the $d \times d$ Jacobian matrices of the corresponding vector functions evaluated at $\mathbf{s}(t)$, and ξ is a parameter characterizing the global coupling strength, which can be set to unity for convenience. We denote the eigenvalues of the coupling matrix $\mathbf{G}$ as $\mu_1, \mu_2, \ldots, \mu_N$ and the associated eigenvectors as $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_N$. While compressive sensing does not require network connectivity, it is meaningful to explore synchronizability only when the underlying network is a single connected component. Since the network is connected, there is only one zero eigenvalue, so the eigenvalues can be sorted as $0 = \mu_1 < \mu_2 \leq \cdots \leq \mu_N$. We then diagonalize the coupling matrix to a block matrix form composed of all the eigenvectors: $\mathbf{Q} = [\mathbf{e}_1; \mathbf{e}_2; \cdots; \mathbf{e}_N]$, which can be used in the transformation, $\delta\mathbf{x} = \mathbf{Q} \cdot \delta\mathbf{y}$, to bring Eq. (3.1) into the following block-diagonally decoupled form,

$$\frac{d\delta\mathbf{y}_i}{dt} = [\mathbf{DF}(\mathbf{s}) - K_i\mathbf{DH}(\mathbf{s})] \cdot \delta\mathbf{y}_i. \tag{3.2}$$

where $K_i = \xi\mu_i$ $(i = 2, ..., N)$ are the coupling strength in the oscillator network. For each K_i value, the corresponding MSF $\Psi(K)$ is the largest Lyapunov exponent of Eq. (3.2) [100]. If, for all possible values of K_i, the corresponding MSFs are all negative, a small perturbation about the synchronous state will vanish exponentially so that it is stable. Since MSFs do not depend on the specific network topology

but on the coupling parameters, we can first infer the parameters from one set of specific measurements and calculate the MSF for arbitrary K so that the emergence of synchronous behavior can be anticipated. This can be done even when links are added or removed, because of MSF's independence of the network structure.

After the MSF is known, the synchronization behavior of the whole oscillator networks can be assessed. For example, suppose the system is not currently in a synchronous state, but there is a region of K, $K_a < K < K_b$, in which the MSF satisfies $\psi(K) < 0$. We can find a suitable positive coupling strength ξ such that $K_a < \xi\mu_2 < \xi\mu_N < K_b$ so as to drive the system into synchronization. This is because, under the stretching/squeezing effect of ξ, all possible K_i's can be brought into the negative MSF region.

3.3 Examples

To illustrate our method to forecast synchronization, we first choose the Erdős-Rényi (ER) type of homogeneous random network consisting of identical Lorenz oscillators as an example, and then extend to scale-free networks and discrete-time nodal dynamics as well. In fact, similar results have been obtained for other network topologies and different types of nodal dynamics besides the cases presented here.

The classical Lorenz system is given by $[\dot{x}, \dot{y}, \dot{z}] = [\sigma(y-x), x(\rho-z)-y, xy-\beta z]$, where we set $\sigma = 10$, $\rho = 28$, and $\beta = 2$ so that the oscillator is chaotic. Time-series data are generated from 6×10^6 numerical-integration steps with maximum step size of 10^{-4}. The Hénon map system is given by $[x_{t+1}, y_{t+1}] = [1 - ax_t^2 + y_t, bx_t]$, and we set $a = 1.4$ and $b = 0.3$ so that the map exhibits chaotic dynamics, for which time series of length $T_N = 100$ are generated. However, the amount of measurement data used in the compressive-sensing algorithm can be much smaller. Using an adjustable sampling frequency $1/\Delta T$ (or iterative interval T_N), we obtain sparse measurement

data to reconstruct the nodal dynamics, coupling pattern and the network structure. In a typical application, some physical knowledge about the underlying complex networked system may be available. This can in fact help reduce the computational complexity and increase the efficiency and accuracy significantly. For example, in the case of Lorenz-oscillator networks, some preliminary understanding of the system can facilitate the choice of the power-expansion order. To be illustrative, we apply the constraint $l_1 + l_2 + l_3 \leq 4$ on the powers of the components x, y, z so that the number of unknown coefficients can be reduced.

The Jacobian matrix of the Lorenz system is

$$\mathbf{DF} = \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho - z & -1 & -x \\ y & x & -\beta \end{pmatrix}. \tag{3.3}$$

The Jacobian matrix of the coupling function, **DH**, for one specific node component, is a 3×3 matrix with only one nonzero element at the corresponding position determined by the coupling pattern. In order to compute the MSF, we need to reconstruct the network structure, find coupling pattern, and determine the parameters characterizing the nodal dynamics.

3.3.1 Predicting Weighted Networks

Figure 3.1 shows the results of predicting a small *weighted* Lorenz-oscillator network. There are in total 122 terms in the coefficient vector $\mathbf{a}$ for each node, in which the 1st to the 35th terms correspond to nodal dynamics vector $\mathbf{b}_i$ and the rest to the coupling vector $\mathbf{c}_i$ with other nodes. The inferred coupling strengths of node #1 with other nodes is shown in Fig. 3.1(a) where, with respect to the number of power-expansion terms with nonzero coefficient values, the predicted coupling terms with other nodes are marked in Fig. 3.1(b). The network structure with node degrees

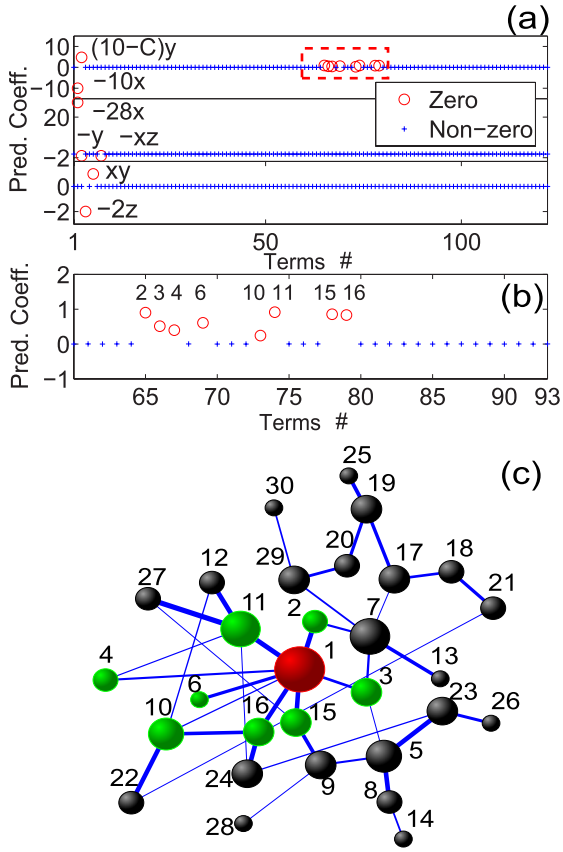

Figure 3.1: Results of detecting dynamical and coupling terms via compressive sensing. The network used is the ER random network with $N = 30$ nodes and connection probability $p = 0.04$. The network is weighted and the symmetric weights are randomly distributed in $[0.1, 1.0]$. Panel (a) shows the prediction results for all three components x, y, z of node #1, where the number of data points (after sampling) used is 70% of the total number of the power-series coefficients assumed. Terms with nonzero coefficients are marked by red circles, while others by blue plus-signs. The first 35 terms are for nodal dynamical equations, and the rest are for the coupling functions. In the first panel for component x, the data points surrounded by the dashed box represent coupling-term coefficients from other node components to component x of node #1, which is magnified in panel (b) with numbers above data points indicating the nodes from which the couplings come. Panel (c) shows the original ER network, where node #1 is highlighted in red, its nearest neighbors are presented in green, and the thickness of the edges indicates the corresponding coupling strength. One-to-one correspondence can be identified between the predicted coefficients in panel (b) and the coupling strengths in panel (c) for each of node #1's neighbors.

and link weights is shown in Fig. 3.1(c). We see that all existent couplings have been successfully predicted, together with the corresponding link weights. Results of prediction of all 122 terms in $\mathbf{a}$ for all three variables x, y, z in the coupled Lorenz-oscillator network are presented in Fig. 3.1(a). Besides the nonzero coupling terms, other nonzero terms represent various power-series terms in the nodal dynamics in each variable. The related forms of the nonzero terms are remarked. For example, $10y-Cy$ is in fact a combination of nodal function and coupling. Based on the indices of the coupling terms, we can identify that the couplings are from y to x, because of the term $\sum_{j=1}^{N} G_{ij}(y_j - y_i)$ in the equation of $\dot{x}$. Therefore, the term $-Cy$ comes from $\sum -G_{ij}y_i$, which has been merged into the nodal dynamical equation. Since all coupling terms are successfully identified, the $-Cy$ term can be separated from the combination, resulting in complete prediction of all power-series terms in the velocity field and coupling function associated with node #1. We have also examined the prediction results for all other nodes in the network and found excellent agreement between the predicted and actual power-series terms governing the whole networked dynamical system.

The efficiency of our method for reconstructing weighted networks can be assessed by addressing the issue of data requirement and sampling frequency when nearly perfect prediction accuracy is achieved. It is useful then to define prediction errors in the coefficient vector $\mathbf{a}$. Since $\mathbf{a}$ is sparse, i.e., most of its elements are zero, it is necessary to calculate the errors for nonzero (existing) and zero (non-existing) terms separately. In particular, the relative error of a nonzero term, E_{term}, is defined as the ratio to the true value of the absolute difference between the inferred and the true values. The prediction error E_{nz} of all nonzero terms in a component,

$$E_{nz} \equiv \langle E_{term} \rangle,$$

is the average over them. For a zero term, a relative error cannot be defined. As an alternative, we define the absolute error as the average value of the inferred zero terms. The prediction errors can then be computed as functions of the amount R_m of measurements, normalized by the total number of unknown coefficients to be determined, i.e.,

$$R_m \equiv \frac{\#\text{ of measurements}}{\#\text{ of all unknown coefficients}},$$

and the sampling time interval ΔT, as shown in Fig. 3.2, where ΔT is the average time interval between two pairs of data points, with each pair containing two nearby data points for the purpose of estimating the corresponding derivative. In Fig. 3.2(a), we see that, for sufficiently large values of R_m, E_{nz} reduces essentially to zero with extremely small error bars, indicating accurate reconstruction of both nodal dynamics and network structures with complete information about the locations of the links and their weights. From Fig. 3.2(b), we observe that a larger sampling interval ΔT tends to facilitate prediction. This can be intuitively understood by noting that suitably large ΔT values weaken the correlation between two adjacency data points, from which reconstruction may be benefited. In both Figs. 3.2(a) and 3.2(b), the Y component appears to be the most difficult one to be fully reconstructed, as the required data amount is the largest. This is due to the presence of the ρx term in the Y component, where the value of the coefficient ρ is much larger than other nodal dynamical and coupling coefficients, requiring more measurements and larger sampling intervals. Our experience indicates that, in general, the data requirement for equations that involve relatively larger coefficients tends to be higher.

In order to assess the accuracy of the predicted weighted network, it is necessary to reconstruct the adjacency matrix for any given coupling scheme. With all expansion coefficients obtained from compressive sensing for all dynamical variables of each oscillator, we can readily form the matrix by using the terms associated with the various

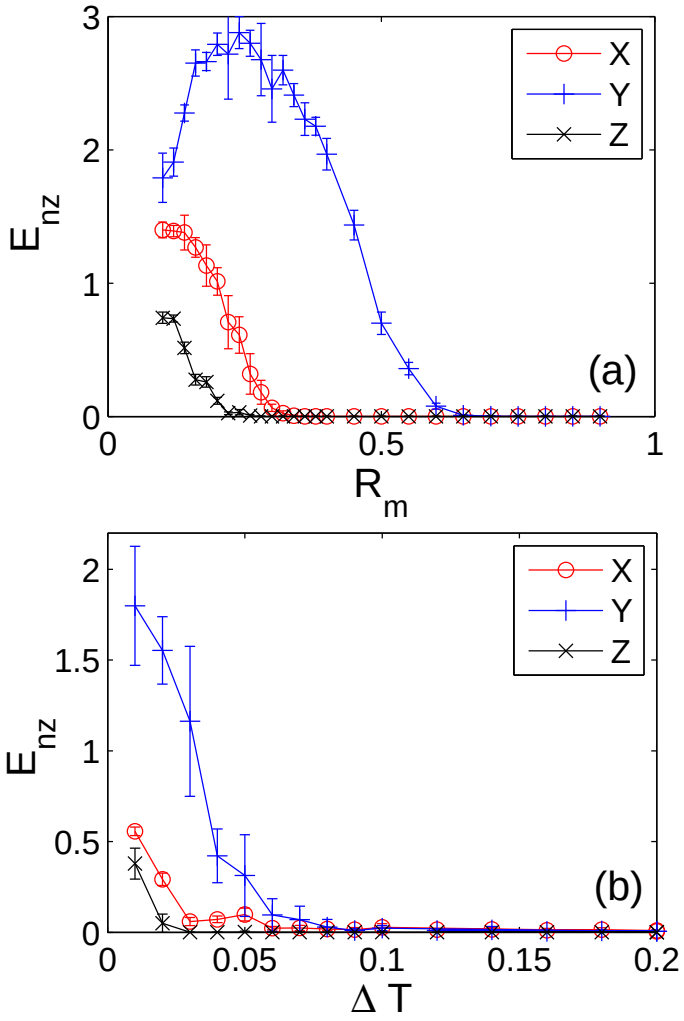

Figure 3.2: Prediction errors as functions of the normalized amount of measurement R_m and sampling interval ΔT for a random symmetric weighted network of $N = 60$ Lorenz oscillators, where the connection probability is $p = 0.04$ and the weights are randomly distributed in $[0.1, 1.0]$. There are possibilities that the generated networks are disconnected, but in order to be able to consider synchronizability, we disregard rare cases where the networks generated consist of isolated components. In (a), the sampling interval is fixed at $\Delta T = 0.1$, whereas in (b), the amount of measurement is fixed at $R_m = 0.6$. In both panels, E_{nz} is averaged over 10 independent network realizations.

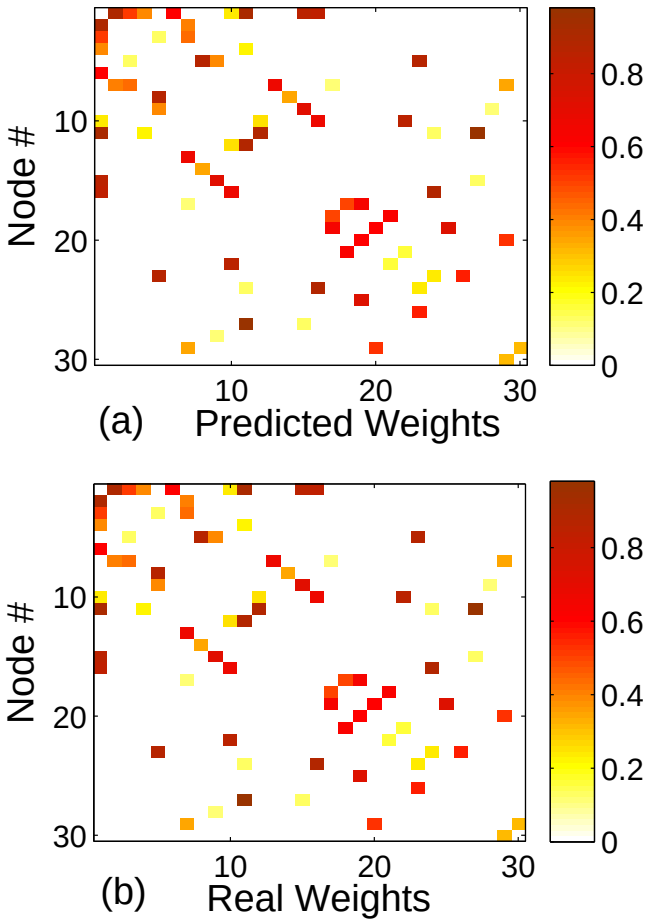

Figure 3.3: Comparison of the reconstructed (a) and the original (b) adjacency matrices for the weighted network shown in Fig. 3.1(c). The coupling scheme is $y \to x$ and the normalized amount of measurements is $R_m = 0.3$.

coupling functions. For example, coupling coefficients from each node contribute to a single row of the adjacency matrix, given any coupling scheme. Figure 3.3 shows, for the $y \to x$ coupling scheme, the reconstructed and the original adjacency matrices. The good agreement between the two suggests that, not only have the link locations been predicted, but also the *values of the corresponding weights*.

To further address the practically important issue of data requirement in reconstructing weighted networks, we define a quantity R_c, which is the critical amount of data required for the prediction error E_{nz} to fall below some predefined small

threshold value (e.g., 0.01), namely,

$$R_c \equiv \inf\{R_m : E_{nz}(R_m) \leq 0.01\}.$$

Although R_c depends on the choice of the threshold, the qualitative behavior of R_c is insensitive to the network structure. For example, we can calculate R_c for different ratios R_{nz} defined as

$$R_{nz} \equiv \frac{\text{\# of nonzero coefficients}}{\text{\# of all unknown coefficients}},$$

where R_{nz} can be adjusted by varying the network size while keeping the average degree unchanged. Figure 3.4(a) shows R_c versus R_{nz} for different ER random networks. We see that, as R_{nz} becomes smaller so that the network becomes more sparse, the value of R_c tends to decrease, indicating that smaller amount of data is required to achieve the same prediction accuracy. This is due to the merit of our compressive-sensing based method in dealing with large networks, i.e., low data requirement. This feature does not depend on the network topology either, as shown in Fig. 3.4(b) for scale-free networks, where R_c is shown as a function of α, the power-law exponent in the degree distribution. When the network size and the average degree are fixed, a smaller value of α corresponds to a more heterogeneous network structure. In this case, the value of R_c is relatively large. The reason is that for a more heterogeneous network, the probability of having dense sets of coefficients for the hub nodes is larger, requiring more data. As α is increased so that the network becomes less heterogeneous, R_c can be reduced.

Eigenvalues of the network coupling matrix can be calculated upon determining the structural parameters of the network. It is thus useful to define another quantity to characterize the accuracy of the reconstructed weighted network. Specifically, we first define the eigenvalue interval that contains all the original eigenvalues as

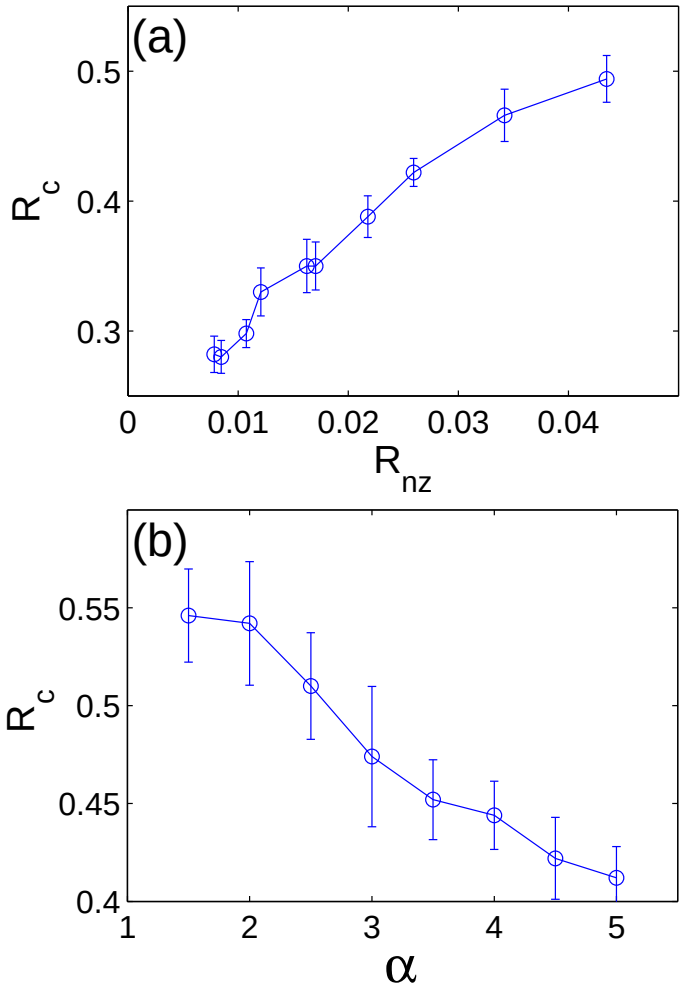

Figure 3.4: (a) For ER random networks, measure of critical data requirement R_c as a function of the density of nonzero coefficients R_{nz}, where R_{nz} is adjusted by fixing the average degree at $k = 3$ and increasing the network size from $N = 20$ to $N = 200$. (b) For scale-free networks, R_c as a function of the power-law exponent α in the degree distribution $p(k) \sim k^{-\alpha}$. The network size is $N = 60$ with the minimal degree $k_{min} = 3$. For both panels, the data points are results of averaging over 10 different network realizations.

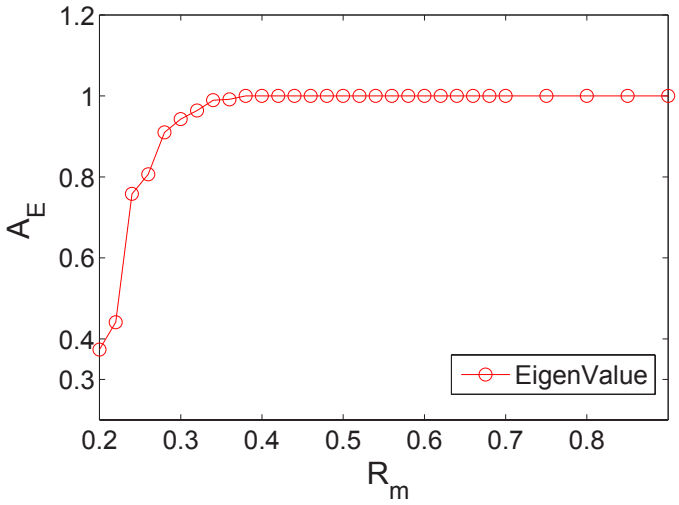

Figure 3.5: For the random Lorenz network under the coupling scheme $y \to x$, accuracy measure A_E of the eigenvalue spectrum of the reconstructed network coupling matrix as a function of the normalized data amount R_m.

$R'_t = (K_2, K_N)$ and the predicted one as $R'_p = (K'_2, K'_N)$. We then define the following quantity A_E to characterize the accuracy of the reconstructed eigenvalue spectrum:

$$A_E = \frac{R'_p \bigcap R'_t}{R'_p \bigcup R'_t} = \frac{\min(K_N, K'_N) - \max(K_2, K'_2)}{\max(K_N, K'_N) - \min(K_2, K'_2)}. \tag{3.4}$$

Here we use a continuous region instead of a set of individual eigenvalues of the coupling matrix for the definition of the true region R'_t, because the necessary condition for the system to be synchronizable is that all eigenvalues must be located in the negative region of MSF $\Psi(K)$. Since the MSF is not involved in the definition of A_E, a convenient choice is to compare the region from the minimum nonzero eigenvalue K_2 to the maximum K_N, which limits our discussion within the systems possessing the type of MSF [see, e.g., Fig. 3.9(b)]. A representative plot of A_E as a function of R_m is shown in Fig. 3.5. We see that, the eigenvalue spectrum can be predicted accurately when R_m exceeds about 35%, due to the low data requirement of compressive sensing.

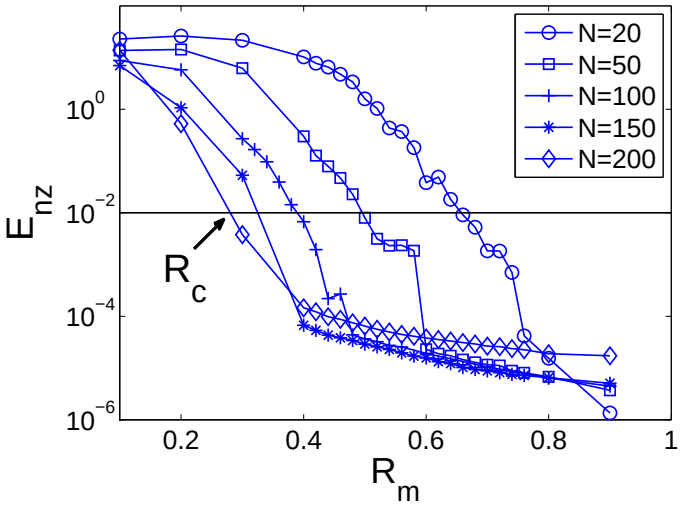

Figure 3.6: For weighted, random networks of Hénon maps, prediction errors as functions of the normalized data amount R_M. The network size varies from 20 to 200, and all the networks tested have the same connection probability $p = 0.04$ with weights distributed in $[5 \times 10^{-4}, 10^{-3}]$. Each point is the result of averaging over 10 independent network realizations. The black solid line at $E_{NZ} = 0.01$ is used to indicate the critical data requirement R_C for each case.

Similar results are obtained from networks of Hénon map systems. In the following examples we discuss the effect of the network size and noise on system reconstruction, and also the issue of computational time. To be illustrative, we assumed weighted random networks with weights distributed in the range $w_{ij} \in [5 \times 10^{-4}, 10^{-3}]$ (so that dynamical trajectories from the Hénon map do not diverge). The coupling function is chosen to be linear, and it occurs between the x variables among the nodes. Applying the compressive sensing algorithm allows us to infer the nodal dynamics and network topology from the coefficients $\mathbf{a}$.

The performance of our method with respect to different network size is an important issue. As shown in Fig. (3.6), as the data amount R_m is increased, for different network sizes ranging from $N = 20$ to $N = 200$, the normalized predicted errors E_{nz} approach zero, as indicated by the black solid line, suggesting that the system can be reconstructed with high accuracy based on small amount of data, regardless of the

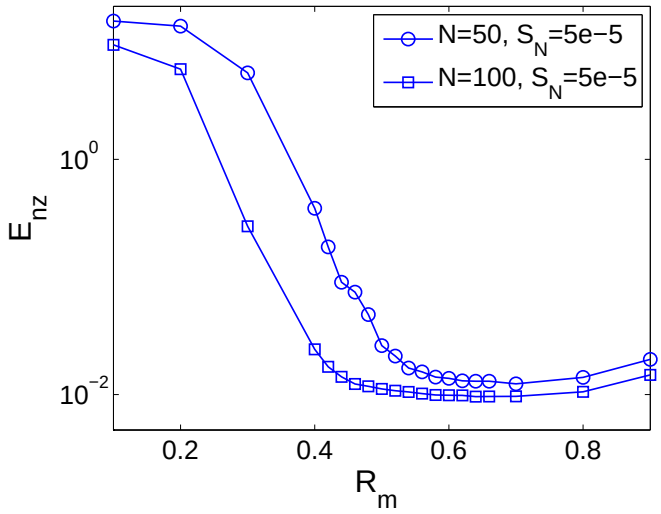

Figure 3.7: For uniform measurement noise, prediction error E_{nz} versus the normalized data amount R_m, where the networks are the same as in Fig. 3.6. Each point is the result of averaging over 10 different network realizations.

network size. While slightly more data are required for larger networks, the amounts are still quite small, i.e., less than the total number of unknown coefficients in the power-series expansion. We also find that the critical data ratio R_c, defined as the relative data amount required to make the normalized predicted error E_{nz} less than a small threshold value (e.g., 0.01), decreases with the network size N. This is in accordance with the results in Fig. 3.4(a), since the degree of sparsity of the unknown vector $\mathbf{a}$ increases with the random network size as the connection probability p is fixed.

Another issue that we have studied is the effect of measurement noise on reconstruction. In our framework, observations of the variable states in one measurement are associated with the state of the system at the particular time, so measurement noise can be quite important. Figure 3.7 shows the reconstruction result when additive noise of amplitude 5×10^{-5} is present. We see that compressive sensing is capable of generating approximate solutions of the networked system even in the presence of

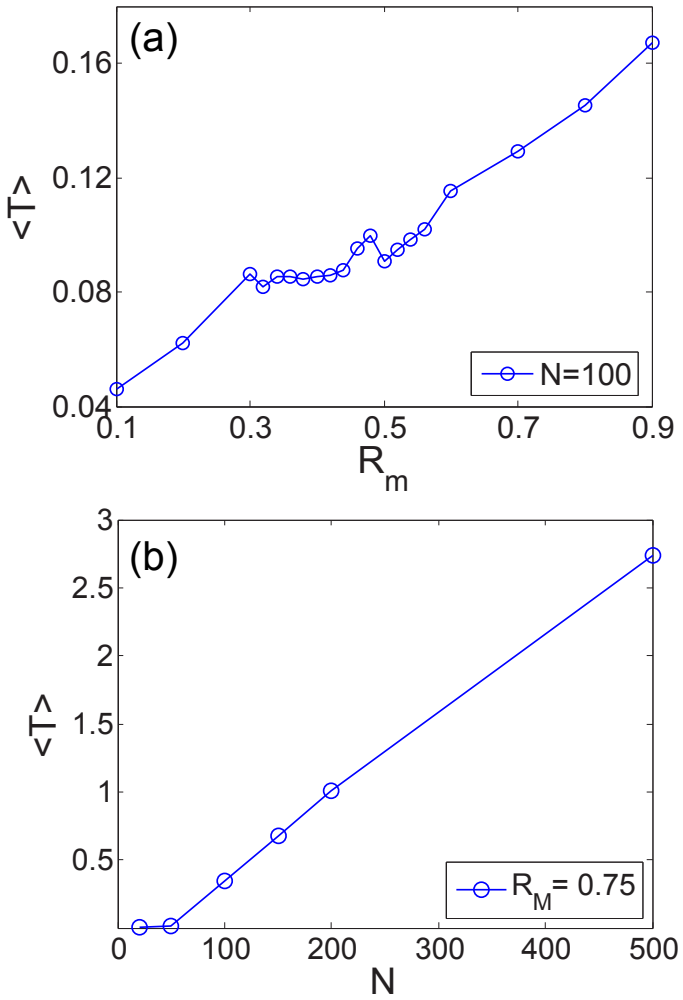

Figure 3.8: For weighted, random networks of Hénon maps, (a) the average computational time T (in an arbitrary unit) required for one variable on one node versus the data ratio R_m, for fixed network size ($N = 100$), (b) T versus the network size N for fixed R_m (0.75) for which accurate reconstruction can be achieved. For both panels, 20 network realizations are used.

noise. The data amount required to reconstruct the network, however, tends to be slightly larger than that in the case where no noise is present.

We have also considered the issue of computational time. In our method, the main computational load lies in solving CS matrix optimization, which depends on the number of unknown coefficients and the number of measurements. We first fix the network sizes at $N = 100$ and record the computation time as the relative data

amount R_m is changed. As shown in Fig. 3.8(a), the required time to reconstruct one coefficient vector (for one variable of one node in the network) scales approximately linearly with the data amount. Next we fix R_m and monitor the required computation time as a function of the network size. For linear coupling, the number of unknown coefficients is proportional to the network size N if it is sufficiently large. We set $R_m = 0.75$ to ensure accurate reconstruction in each case, so the amount of data used for reconstruction increases linearly with the number of unknown coefficients. Figure 3.8(b) shows the result, where the network size varies from $N = 20$ to $N = 500$. We see that the required computation time indeed increases approximately linearly with the number of unknown coefficients.

3.3.2 Prediction of Network Synchronizability from Data

A full reconstruction of nodal dynamics allows us to calculate the MSF Ψ as a function of $K \equiv \xi\mu$ for any given coupling scheme. To be illustrative, we calculate the MSFs for four different coupling schemes ($x \to x$, $y \to x$, $z \to x$, and $z \to z$) for the coupled network of Lorenz oscillators, as shown in Fig. 3.9. These coupling patterns generate distinct behaviors of the MSF in terms of its number of zeros. If a region of $\Psi(K) < 0$ exists, emergence of stable synchronization is likely for the oscillator network, regardless of the network structure; otherwise synchronization is unlikely for any network structure. In Fig. 3.9, for example, for the $x \to x$ coupling scheme, there is a relatively large synchronization region for K beyond a critical value. For the $y \to x$ scheme, a synchronization region exists but its size is not as large as that for the case of $x \to x$ coupling. For the $z \to z$ coupling scheme, there are in fact two separated synchronization regions. In contrast, for the $z \to x$ coupling scheme, synchronization is unlikely because $\Psi(K)$ is positive for all values of K. A more systematic analysis of the MSF behaviors for typical nonlinear oscillators can be found in Ref. [101]. The

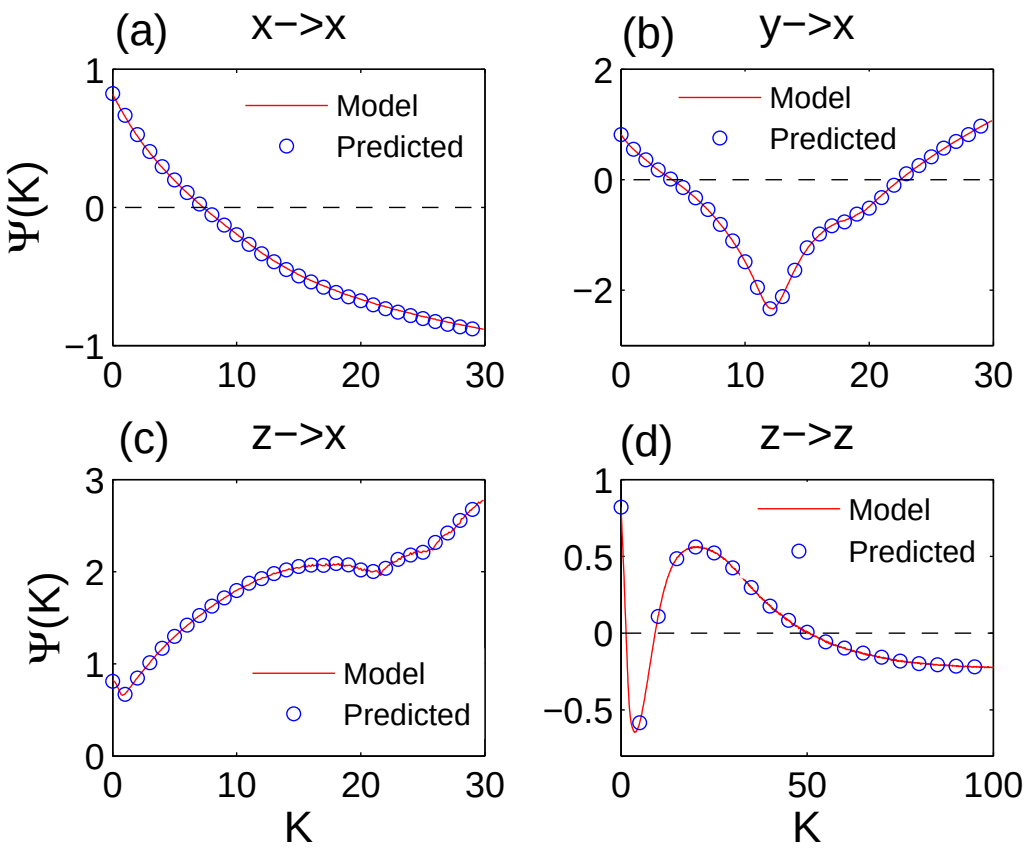

Figure 3.9: Comparison of MSFs calculated from predicted parameters (blue circles) and from real ones (red lines) for the random Lorenz oscillator network. Panels (a-d) are for coupling schemes $x \to x$, $y \to x$, $z \to x$, and $z \to z$, respectively. All time-series data are generated by the same oscillator network as in Fig. (3.2).

excellent agreement between the true and predicted MSFs shown in Fig. 3.9 suggests that our compressive-sensing based approach can lead to quite reliable estimate of the MSF at a quantitative level. Likewise, the boundaries between synchronous and asynchronous regions can also be precisely identified, rendering possible anticipation of the emergence of synchronization in the underlying network system.

To quantify the performance of our method in identifying the synchronization region, we define a measure of agreement, denoted by A_M, between the predicted and true synchronization region, as exemplified in Fig. 3.9(b) for the $y \to x$ coupling scheme. Specifically, we denote the true synchronization region R_t by (K_a, K_b) in

which the MSF is negative, and denote the predicted region R_p by (K'_a, K'_b). We thus define

$$A_M = \frac{R_p \bigcap R_t}{R_p \bigcup R_t} = \frac{\min(K_b, K'_b) - \max(K_a, K'_a)}{\max(K_b, K'_b) - \min(K_a, K'_a)}, \tag{3.5}$$

where generally $A_M \leq 1$. Two extreme cases are $A_M = 0$ when $R_p \bigcap R_t = \emptyset$, and $A_M = 1$ when $R_p = R_t$, which indicate perfect prediction. Results are shown in Fig. 3.10, where A_M approaches unity as the amount of measurement exceeds only about 65% of the number of assumed coefficients to be predicted. For the case of single intersection K_a of MSF with $\Psi(K) = 0$, as shown in Fig. 3.9(a) for the $x \to x$ coupling scheme, we can define an agreement measure in a similar way:

$$A_M = \frac{\min(K_a, K'_a)}{\max(K_a, K'_a)}, \tag{3.6}$$

where $0 \leq A_M \leq 1$. In cases where there are multiple synchronization regions, e.g., as happened for the $z \to z$ coupling scheme in Fig. 3.9 (d), the agreement measure can be taken as the average of all measures, one calculated from each separate region.

3.4 Data-based Anticipation and Control of Network Synchronization

Based on the reconstructed network structure and dynamics, we now propose a strategy to anticipate and control collective dynamics of complex oscillator networks. The base of control is prediction of future behavior by decoding the available time series at the present. If the natural dynamics in the future are undesirable, one can implement certain control scheme to drive the system to avoid the undesirable state before it occurs. This, however, requires relatively complete knowledge about the networked dynamical system which, as we have demonstrated in Sec. 3.3, can be achieved by exploiting the compressive-sensing paradigm.

To be concrete, we discuss the case where synchronization is a desirable state of operation for the system, assuming that the system is not synchronized at the

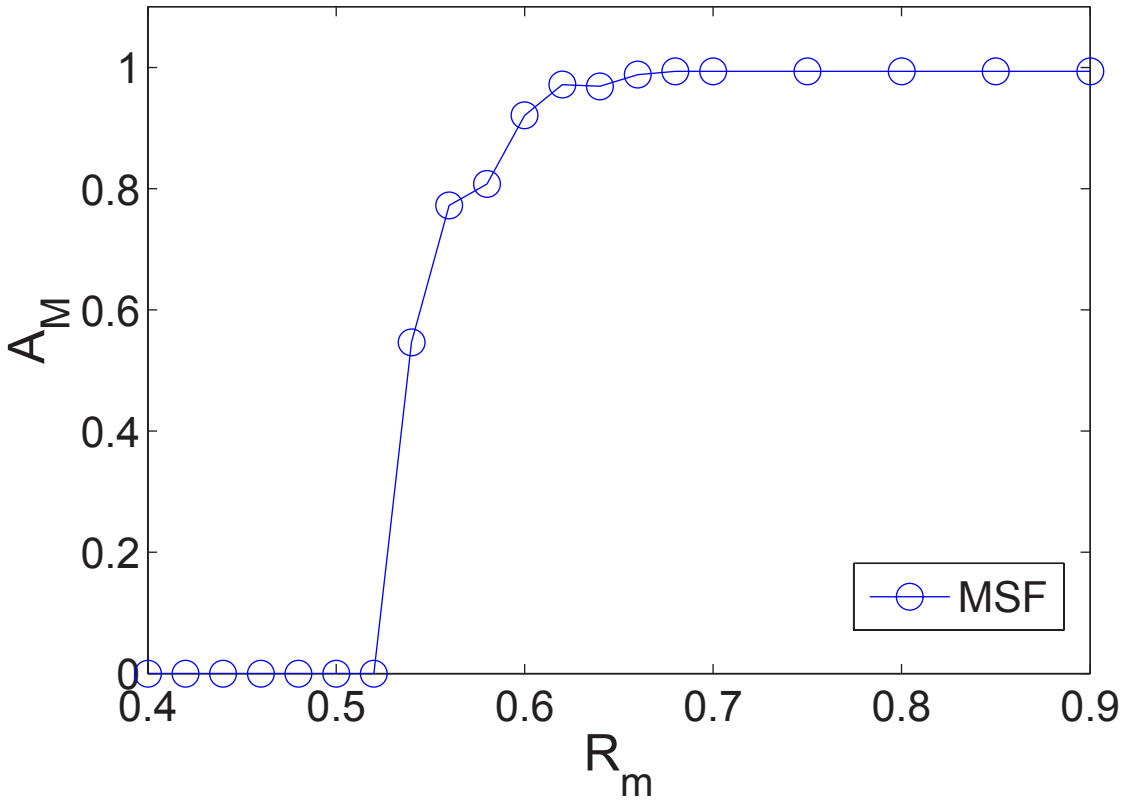

Figure 3.10: Measure of agreement of synchronization prediction A_M as a function of R_m for the MSF shown in Fig. 3.9(b), where the coupling scheme is $y \to x$.

present. The first step is to determine, from currently available time series, whether synchronization is intrinsically likely to emerge. An answer can be obtained by using the reconstructed network structure and dynamics to estimate the network eigenvalue spectrum and MSF. The answer can be affirmative, for example, if the MSF is predicted to be negative in an open generalized coupling-parameter interval. That the system is not currently synchronized indicates that the normalized eigenvalue spectrum does not fall into the interval and, hence, suitable control can be applied to rescale and shift the eigenvalue spectrum into the negative MSF interval. To illustrate this method, we use the network system of coupled chaotic Lorenz oscillators in Sec. 3.3. Figure 3.11(a) shows some representative time series in a case where

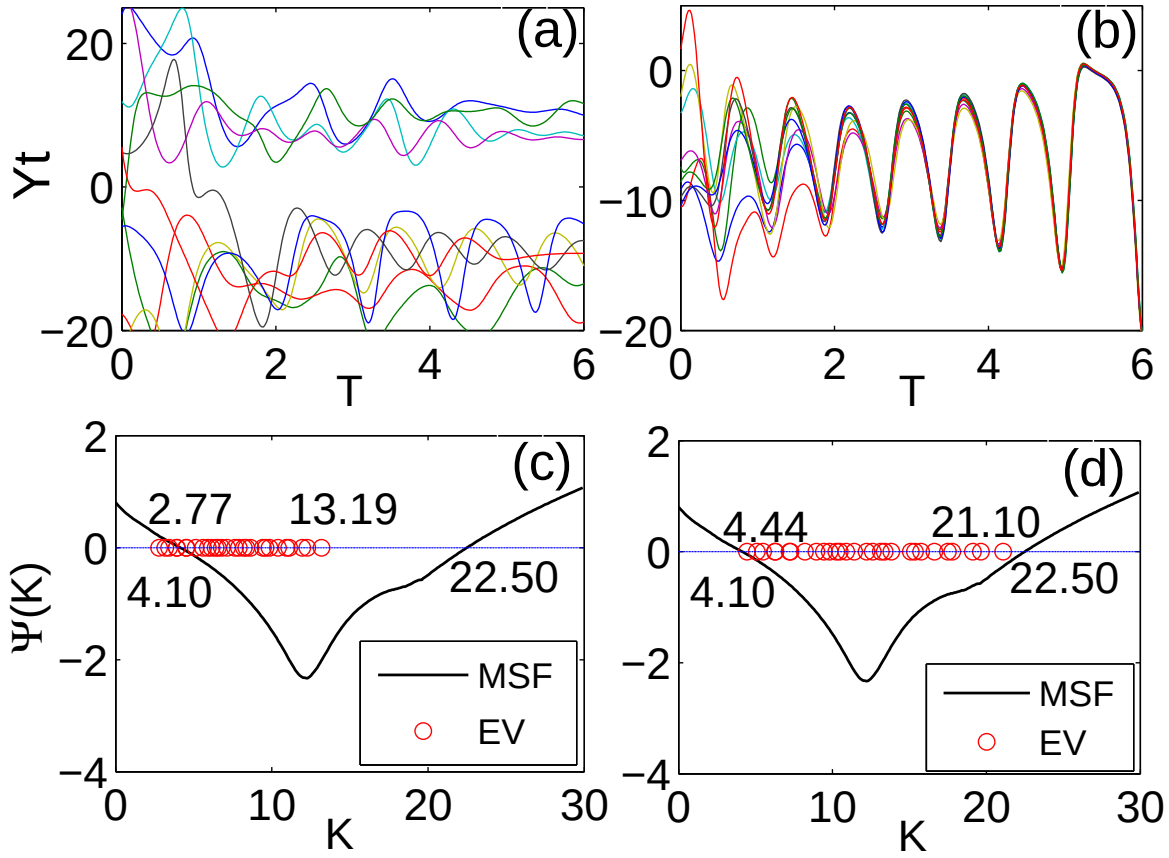

Figure 3.11: (a,b) Time series of y component for 10 of the $N = 30$ nodes in two random networks of global coupling strength $\xi = 1$ and $\xi = 1.6$, respectively. The network is not synchronized in (a) but there is synchronization in (b). Other parameters are the same for both bases: connection probability $p = 0.2$ and the weight distribution interval is $[0.9, 1.0]$. (c,d) Rescaled eigenvalues $K_i (= \xi\mu_i)$ (denoted by red circles) of the network coupling matrices with respect to the MSF (denoted by black solid lines) inferred from the same nodal dynamics and coupling scheme from the time series in (a,b), respectively.

the network is not synchronized, and the corresponding MSF and eigenvalue spectrum calculated from the reconstructed network structure and dynamics are shown in Figure. 3.11(c). We see that some values of K [data points in Figure. 3.11(c)], the product between the coupling strength ξ and eigenvalues μ, are not located in the synchronizable region as indicated by the MSF [curve in Figure. 3.11(c)]. Thus, at the current parameter setting, synchronization cannot be realized in the system. In order for synchronization to emerge, all K values must fall into a region where the MSF is negative. A simple and practical way to manipulate K is to adjust the coupling strength but to keep the nodal dynamics and network structure unchanged. When the coupling strength ξ is modified, the network system can indeed achieve synchronization, as shown by the synchronous time times in Figure. 3.11(b). Examination of the MSF and eigenvalue spectrum indicates that, indeed, in this case all K values fall into the negative MSF interval. We stress that a prerequisite to this simple control scheme is full knowledge of the network structure and dynamics which, as we have demonstrated, can be faithfully reconstructed based solely on small amount of data.

3.5 Conclusion and Discussion

Reconstructing dynamical systems based on time series is a problem of significant interest with broad applications in many areas of science and engineering. However, this problem has been outstanding in nonlinear dynamics because, despite previous efforts [65] in phase-space reconstruction using the standard delay-coordinate embedding method [63] to decode the topological properties of the underlying system, how to accurately infer the underlying *nonlinear system equations* remains largely an unsolved problem. In principle, a nonlinear system can be approximated by a large collection of linear equations in different regions of the phase space, which can indeed

be achieved by reconstructing the Jacobian matrices on a proper grid that covers the phase-space region of interest [108, 109]. However, the accuracy and robustness of the procedure are challenging issues, which include the difficulty associated with the required computations. The recently emerged paradigm of compressive sensing [22, 23, 24, 25, 26] provides a possible approach to addressing the dynamical-system reconstruction problem [29, 30]. In particular, to be able to fully reconstruct dynamical systems using only time series data is based on the fact that the dynamics of natural and man-made systems are determined by smooth enough functions that can be approximated by finite expansions. The major task then becomes estimating the coefficients in the series representation of the vector field governing the system dynamics, for example, from a power-series expansion. In general, the power series can contain high-order terms, and the total number of coefficients to be estimated can therefore be quite large. This is a very difficult problem to solve, since large amounts of data would be needed, making the computations extremely demanding. However, most of these coefficients are either zero (or negligible), rendering sparse the vector of coefficients and applicable of the compressive-sensing paradigm.

The main achievement of this paper is to extend our recently developed method of reconstructing dynamical systems [29, 30] to complex *weighted* oscillator networks and then to address the problem of forecasting collective dynamics. In general, to predict the emergence of collective dynamics is an extremely difficult problem, and it is necessary to focus on a relatively well understood type of collective dynamics. We choose synchronization. We have detailed the basic principle of time-series based prediction of synchronization in complex oscillator networks. We have also demonstrated, using a prototype of oscillator networks with non-uniform coupling strengths (so that the network is weighted), that our compressive-sensing approach can indeed fully reconstruct the network structure and dynamics, based on which the

emergence of synchronous dynamics can be anticipated. We have also articulated and demonstrated a method, based on full reconstruction of complex networked dynamical system that is not yet synchronized, to make it synchronizable by parameter adjustment.

We emphasize that a full reconstruction of a complex oscillator networked system from time series is possible only when the system is not in synchronization, and the information can then be used to forecast or anticipate synchronization in the future. If the system is already synchronized, time series from different nodes are practically identical so that it is not possible to reconstruct the network structure. However, there may exist a solution to this problem. In particular, given a network system that is already synchronized, we hypothesize using small, random, and rare perturbations to disturb the system so that it desynchronizes temporally. Since the synchronization state is stable, the system will settle back to being synchronous quickly. However, the window of temporal desynchronization provides us with an opportunity to probe the system structure. While the transient desynchronization phase may be short, our compressive-sensing method can be particularly suitable because of the extremely low data requirement.

Chapter 4

DETECTING HIDDEN NODES IN COMPLEX NETWORKS FROM TIME SERIES

4.1 Definition of Hidden Node Detection

The power of science lies in its ability to infer and predict the existence of objects from which no direct information can be obtained experimentally or observationally. A well known example is to ascertain the existence of black holes of various masses in different parts of the universe from indirect evidence, such as X-ray emissions. In the field of complex networks, the problem of detecting *hidden* nodes can be stated, as follows. Consider a network whose topology is completely unknown but whose nodes consist of two types: one accessible and another inaccessible from the outside world. The accessible nodes can be observed or monitored, and we assume that time series are available from each node in this group. The inaccessible nodes are shielded from the outside and they are essentially "hidden." The question is, can we infer, based solely on the available time series from the accessible nodes, the existence and locations of the hidden nodes? Since no data from the hidden nodes are available, nor can they be observed directly, they act as some sort of "black box" from the outside world. Despite recent works on uncovering network topologies [16, 17, 18, 20, 21, 29, 30], to our knowledge, the problem of detecting hidden nodes in complex networks has not been addressed. Solution of the problem, however, has potential applications in different fields of significant current interest. For example, to uncover the topology of a terrorist organization and especially, various ring leaders of the network is a critical task in defense. The leaders may be hidden in the sense that no direct information

about them can be obtained, yet they may rely on a number of couriers to operate, which are often subject to surveillance. Similar situations arise in epidemiology, where the original carrier of a virus may be hidden, or in a biology network where one wishes to detect the most influential node from which no direct observation can be made.

In this paper, we present a completely data-driven, compressive-sensing based [22, 31, 25, 26, 24] approach to inferring the existence and locations of hidden nodes in complex networks. The general principle underlying our method can be understood by referring to Fig. 4.1(a) where, for illustrative purpose, a network of 20 nodes with directed interactions is shown. Suppose nodes No. $1-19$ are accessible to the external world, while node No. 20 (in gray) is hidden and thus inaccessible from the outside. The hidden node has two neighbors: No. 9 and No. 18 (in green), and the remaining 17 nodes are marked as red. Every red node thus has the property that time series from itself and *all* its neighbors are available, but for each green node, although time series from itself is available, the same is not true for all its neighbors due to its link with the hidden node. Generally, the time series can be regarded as being generated by the combination of nodal and coupling dynamics, and one wishes to base on the time series to predict the various dynamical equations so that the dynamical processes on various nodes and the network topology can be uncovered. As we shall demonstrate, for a given node, this can indeed be achieved provided that time series from the node and all its neighbors are available. Referring to Fig. 4.1(a), this means that the dynamical equations and the links from/to all red nodes can be predicted. However, significant errors would arise in the prediction of the green nodes due to incompleteness of information about their neighbors. By examining the prediction errors of all accessible nodes, the ones that are connected to the hidden node will then show anomalies, providing a way to infer its existence and location (e.g., connected to the two green nodes in Fig. 4.1(a).

4.2 Detect Hidden Node(s) Using Compressive Sensing

The paradigm of compressive sensing [22, 31, 25, 26, 24] aims to reconstruct a sparse vector $\mathbf{a} \in \mathbb{R}^N$ from linear measurements $\mathbf{M}$ in the form $\mathbf{M} = \mathbf{G} \cdot \mathbf{a}$, where $\mathbf{M} \in \mathbb{R}^K$ and $\mathbf{G}$ is an $K \times N$ matrix. The compressive sensing theory [22, 31, 25, 26, 24] guarantees that, when most components in the unknown vector $\mathbf{a}$ are zero, it can be reconstructed by fewer measurements than the number of components. The unknown vector $\mathbf{a}$ can be solved, for example, by a convex optimization procedure based on L_1 norm. Our recent work [29, 30] demonstrated that the problem of data-based network reconstruction can be casted into the form of $\mathbf{M} = \mathbf{G} \cdot \mathbf{a}$.

We consider networked systems for which the nodal dynamics, described by the vector function $\mathbf{F}_i(\mathbf{x}_i)$, can be separated from the interactions or coupling with other nodes in the network, mathematically described by the coupling function $\mathbf{H}_{ij}(\mathbf{x}_i, \mathbf{x}_j)$. The system can then be written as $\mathbf{M}_i = \mathbf{F}_i(\mathbf{x}_i) + \sum_{j \neq i}^{N} w_{ij} \mathbf{H}_{ij}(\mathbf{x}_i, \mathbf{x}_j)$, where $\mathbf{M}_i$ is the system response, either in discrete or continuous time. For example, for discrete-time mapping system, $\mathbf{M}_i$ are the state variables at the next time step, while in continuous system $\mathbf{M}_i$ are the derivatives of the corresponding variables. To illustrate our method to detect hidden nodes in a concrete manner, we assume that the nodal and coupling functions can be written as some series expansion, e.g., power or Fourier series. In particular, we write: $\mathbf{F}_i(\mathbf{x}_i) = \sum_\gamma \tilde{a}_i^{(\gamma)} \tilde{g}_i^{(\gamma)}(\mathbf{x}_i)$ and $\mathbf{H}_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \sum_\beta a_{ij}^{(\beta)} g_{ij}^{(\beta)}(\mathbf{x}_i, \mathbf{x}_j)$, where $\tilde{g}_i^{(\gamma)}$ are the expansion bases associated with $\mathbf{x}_i$ only, and $g_{ij}^{(\beta)}$ are with respect to both $\mathbf{x}_i$ and $\mathbf{x}_j$. Next we combine the bases $\tilde{\mathbf{g}}_i(t)$ and $\mathbf{g}_{ij}(t)$ at time t into a row vector, and the coefficients $\mathbf{a}_i^{(\alpha)}$ and $\mathbf{a}_{ij}^{(\beta)}$ into a constant column vector. The time-series vector of responses $\mathbf{M}_i(t)$ for node i can then be expressed by the product of

the matrix $\mathbf{G}_i$ and the *to-be-determined* coefficient vector $\mathbf{a}_i$, with $\mathbf{G}_i$ given by

$$\mathbf{G}_i = \begin{pmatrix} \tilde{\mathbf{g}}_i(t_1) & \mathbf{g}_{i1}(t_1) & \cdots & \mathbf{g}_{ij}(t_1) & \cdots & \mathbf{g}_{iN}(t_1) \\ \tilde{\mathbf{g}}_i(t_2) & \mathbf{g}_{i1}(t_2) & \cdots & \mathbf{g}_{ij}(t_2) & \cdots & \mathbf{g}_{iN}(t_2) \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ \tilde{\mathbf{g}}_i(t_m) & \mathbf{g}_{i1}(t_m) & \cdots & \mathbf{g}_{ij}(t_m) & \cdots & \mathbf{g}_{iN}(t_m) \end{pmatrix}, \tag{4.1}$$

where $\tilde{\mathbf{g}}_i(t)$ is the set of bases of $\mathbf{F}_i(\mathbf{x}_i)$, and $\mathbf{g}_{ij}(t)$ is the set of expansion bases of $\mathbf{H}_{ij}(\mathbf{x}_i, \mathbf{x}_j)$. Elements in the vector $\mathbf{M}_i(t)$ contain system response $m_i(t)$ at different t. In particular, when the vector $\mathbf{a}_i$ is determined by solving $\mathbf{M} = \mathbf{G} \cdot \mathbf{a}$, the dynamical equations for the set of corresponding variables at all nodes become known. Note that the vector $\mathbf{a}_i$ contains all the coupling weights from other nodes to i as in $\mathbf{g}_{ij}(t)$ and complete information about the nodal dynamical equations as in $\tilde{\mathbf{g}}_i(t)$. Previous works [29, 30] demonstrated that solutions to the compressive sensing problem can be obtained but only when time series from *all* nodes are available, i.e., when there is no hidden object.

To devise a compressive-sensing based methodology for detecting hidden nodes, we consider the case of one hidden node (or one cluster of hidden nodes). Let node i be one of the immediate neighbors of the hidden node. Due to lack of time series from the hidden node, the form $\mathbf{M} = \mathbf{G} \cdot \mathbf{a}$ is violated for node i, despite the available time series from other nodes in the network. That is, due to the missing time series from the hidden node and consequently missing elements in $\mathbf{a}$, it is not possible to obtain the true solution of the dynamical equations of node i. If a node does not neighbor any hidden node, time series from itself and all its direct neighbors are available, rendering valid the form $\mathbf{M} = \mathbf{G} \cdot \mathbf{a}$ for such a node. The practical importance is that the errors in the prediction of the dynamics of the immediate neighbors of the hidden node will be much larger than those associated with nodes that do not have any hidden node in their neighborhoods. The predicted characteristics of all neighboring

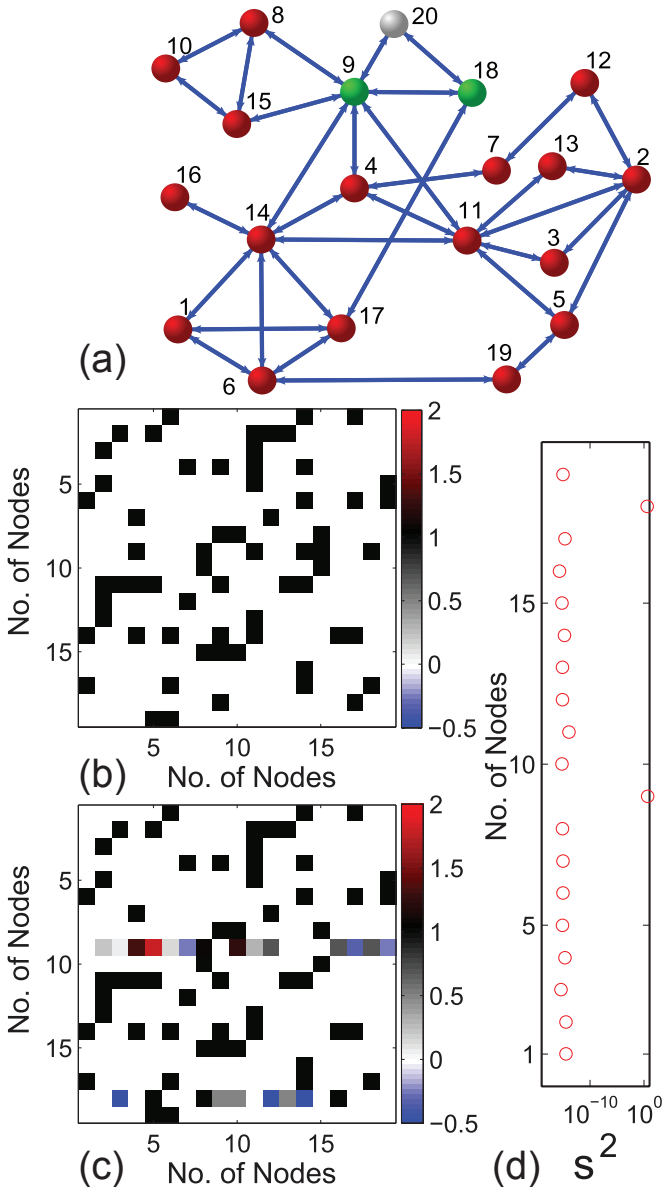

Figure 4.1: (a) Illustration of a complex network with a hidden node. (b) Representation of the true adjacency matrix, (c) reconstructed adjacency matrix elements for nodes except the hidden node based on time series from these nodes. (d) Variance σ^2 of the reconstructed coefficient vector **a** for all nodes, calculated by using 10 different random segments from the available experimental time series. The variances of the two green nodes (No. 9 and No. 18) are much larger than those of the red nodes, indicating that they are the neighbors of the hidden node.

nodes of the hidden node will then show significant anomalies as compared with those of other nodes. The anomalies can then be used to identify all nearest neighbors of the hidden node, which in turn imply its existence and its position in the network.

While our general idea of detecting hidden nodes can be formulated using different types of dynamical systems, to be concrete we describe how this can be done using evolutionary-game type of dynamics. Such dynamical processes can be used to model generic agent-to-agent interactions in economical, social, or even certain biological networks [110, 111]. In an evolutionary-game system, the neighbors of the hidden node can be identified by utilizing the stability criterion with respect to different measurements. More specifically, in an evolutionary-game system, at any time a player can take on one of two strategies: cooperation (C) or defection (D), mathematically represented as $\mathbf{S}(C) = (1, 0)^T$ and $\mathbf{S}(D) = (0, 1)^T$, respectively. The payoffs of the two players in a game are determined by their strategies and the payoff matrix $\mathbf{P}$. For example, for the classical prisoner's dilemma game (PDG), the matrix elements are $P_{11} = 0$, $P_{12} = 0$, $P_{21} = b$, and $P_{22} = 0$, where $1 < b < 2$ is a parameter characterizing the temptation to defect. At each time step, all agents in the network play the game with their neighbors simultaneously and gain rewards. For agent i, the reward is $m_i = \sum_j a_{ij}\mathbf{S}_i^T\mathbf{P}\mathbf{S}_j$, where $\mathbf{S}_i$ and $\mathbf{S}_j$ denote the strategies of agents i and j taken at the time and a_{ij} is the coupling strength between them. After obtaining its payoff, an agent updates its strategy according to its own and neighbors' payoffs, attempting to maximize its payoff at the next round. We assume that the strategy and payoff data of agents are available except those of the hidden node. In particular, we choose $\mathbf{g}_{ij}(t) = \mathbf{S}_i^T(t) \cdot \mathbf{P} \cdot \mathbf{S}_j(t)$ and ignore $\tilde{\mathbf{g}}_i$, the payoff of node i at different time t can be expressed as $\mathbf{M}_i(t) = \mathbf{G}_i \cdot \mathbf{a}_i$, where $\mathbf{G}_i$ is to be constructed as specified in Eq. (4.1), and the vector $\mathbf{a}_i$ to be determined contains all interaction strength

between nodes i and other accessible nodes in the network. The network structure is uncovered after $\mathbf{a}$'s for all nodes are determined.

4.3 Locate Hidden Node(s) in Social Networks

As an example, we present results of experimentally detecting a hidden node from a social network hosting evolutionary-game dynamics. In the experiment, 20 participants from Arizona State University played the prisoner's dilemma game (PDG) iteratively, with a pre-specified payoff parameter. The player with the highest normalized payoff (total payoffs normalized by their degrees) summed over time was the winner. The players can gamble with all their nearest neighbors in the pre-existing social network [Fig. 4.1(a)]. The network was determined by surveying the friendships among those participants, and it exhibits some typical properties of real social network, such as the much larger degree in some hub nodes. During the experiment, the strategies of each player and the gained payoff were recorded in all the 32 rounds, except for the hidden node No. 20. The true adjacency matrix of accessible nodes is represented in Fig. 4.1(b), and the predicted matrix is shown in Fig. 4.1(c). We see that the links of the two neighboring nodes (No. 9 and No. 18) of the hidden node No. 20 cannot be predicted. Especially, the two nodes are predicted to have links with almost all nodes in the network, which is highly unlikely for a random network that is typically sparse. While the predicted loss of sparsity of certain nodes is an indication that they might be in the neighborhood of some hidden node, the condition is not sufficient in general, because of the existence of hub nodes with significantly more links than average in a complex network. Other conditions must then be sought in order to identify the neighbors of the hidden nodes. Our idea is to exploit the stability of the predicted solution with respect to different measurements used for compressive sensing. In particular, for the neighboring nodes of the hidden node,

due to the lack of information needed to solve the underlying compressive-sensing problem, when different segments of the time series are used, the algorithm will yield different coefficient vectors $\mathbf{a}$. However, for a node not in the immediate neighborhood of the hidden node, the predicted vector $\mathbf{a}$ should be the same for different data segments, for the corresponding coefficients with the hidden node are zero. As shown in Fig. 4.1(d), the variances in $\mathbf{a}$ of nodes No. 9 and No. 18 from a number of predictions are much larger than those (essentially zero) of other nodes. Violation of sparsity in combination with the instability of the predicted solution then allows us to identify all neighbors of the hidden node, and consequently itself, with high confidence.

To systematically characterize the accuracy and efficiency of our method to detect hidden nodes, we calculate the prediction error of links of all nodes (except the hidden node and its neighbors) in terms of the amount of required data. For an individual node, the prediction error is defined as the ratio of the absolute difference between the true adjacency matrix elements of all links associated with this node and the predicted elements to the nonzero true element values. The average over all nodes, excluding the neighbors of the hidden node, gives the total prediction error E_{nz}. To explore the effect of network size, we study networks of systematically varying size, ranging from 20 to 200 nodes. Figure 4.2 shows, for networks of 60 nodes and 100 nodes, E_{nz} as functions of the required data, which are the number of measurements N_m normalized by the number of terms $N_{nz} + N_z$ in the unknown vector $\mathbf{a}$. We see that, for the network of 60 nodes, when the measurement ratio exceeds 0.4, E_{nz} is close to zero, demonstrating that 40% data is sufficient to reconstruct the links and detect the location of the hidden node. For the network of 100 nodes, the data requirement is slightly smaller because the unknown coefficient vector is sparser. To further explore the relation between the data requirement and $N_{nz}/(N_{nz} + N_z)$, the sparsity measure

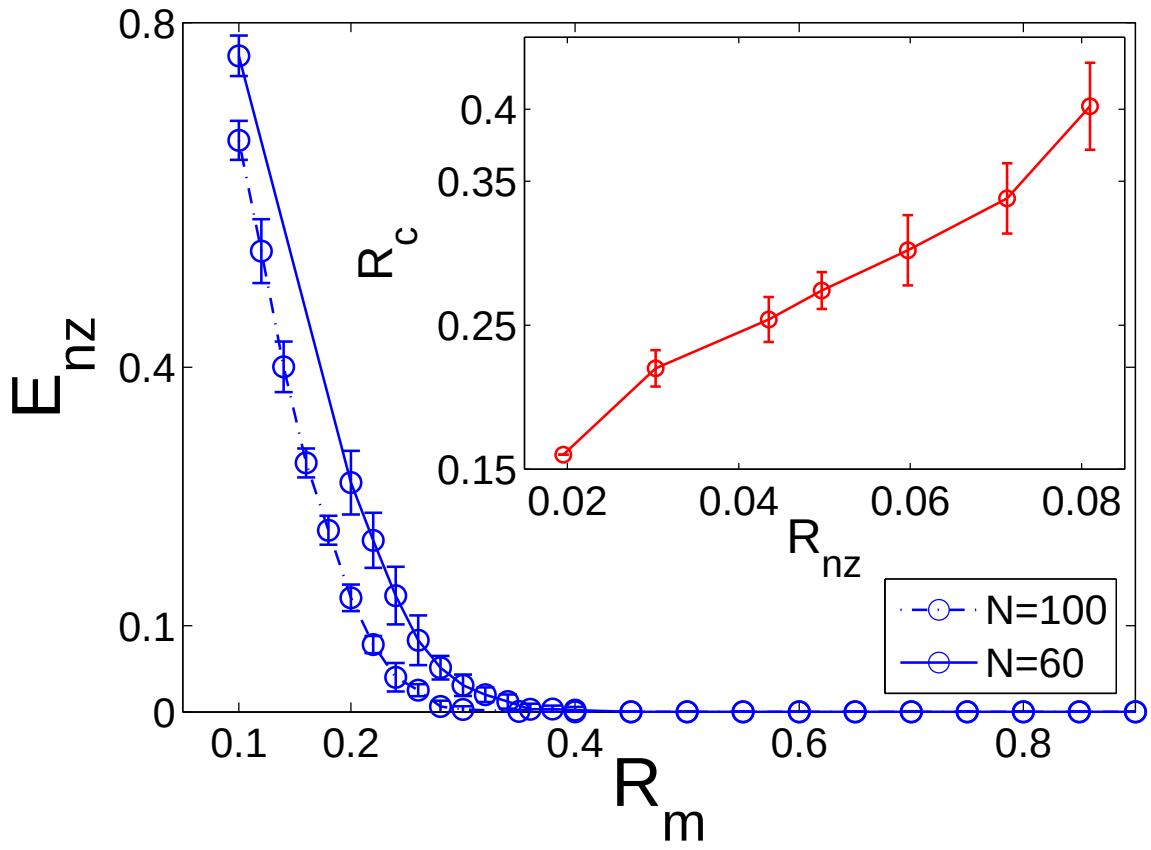

Figure 4.2: For directed, weighted random networks of 60 nodes and 100 nodes, prediction error E_{nz} as a function of the ratio R_m. The ratio R_c as a function of the ratio $R_{nz} \equiv N_{nz}/(N_{nz}+N_z)$ is shown in the inset. The average connecting probability of the network is $p = 0.04$, and the link weights are uniformly distributed between 1 and 6. The error bars are calculated from 20 independent network realizations.

of the vector $\mathbf{a}$ to be predicted, we define a threshold of normalized measurement R_c required for full reconstruction of the network dynamical system when the error E_{nz} is less than 10^{-2}. The sparsity measure can actually be adjusted by varying the network size while keeping the average node degree unchanged. As shown in the inset of Fig. 4.2, we observe that, as $\mathbf{a}$ becomes more sparse, the measurement threshold R_c is reduced accordingly. This also demonstrate the efficiency of our method for different network scales. These results illustrate the power of our compressive-sensing based method to locate hidden nodes with low data requirement.

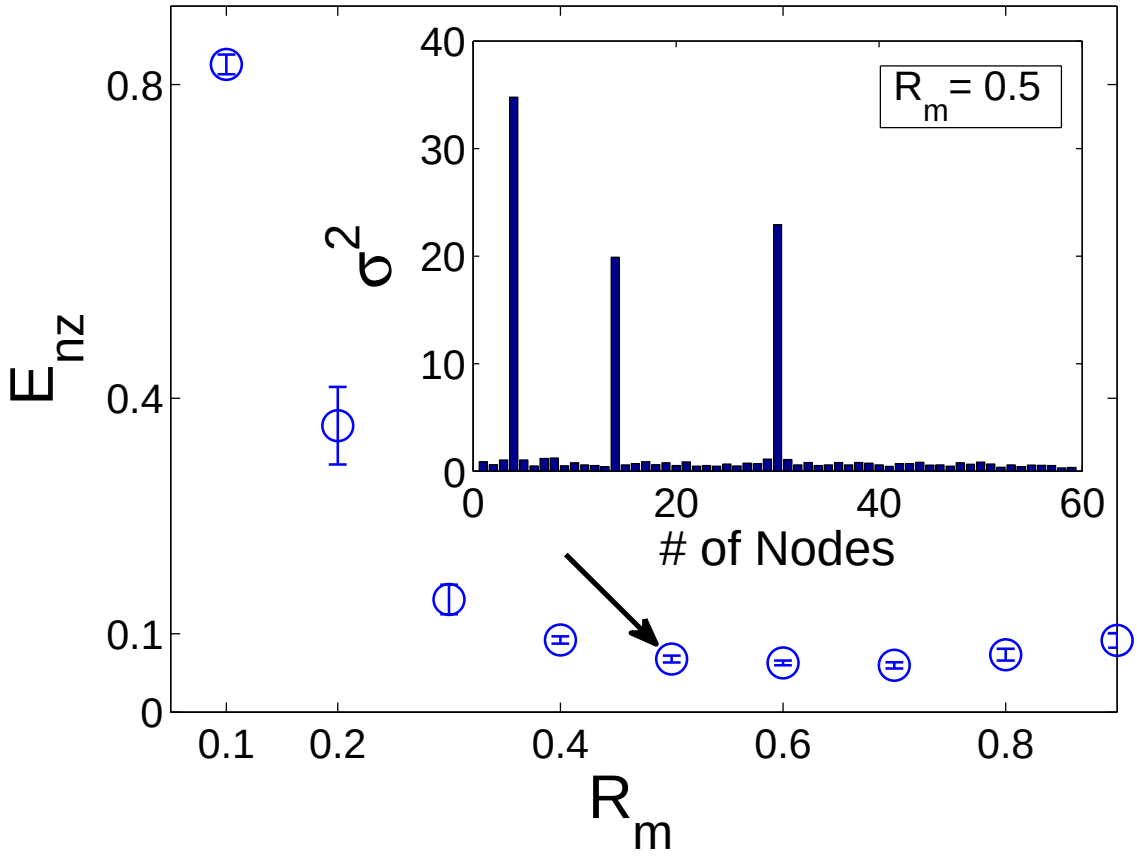

Figure 4.3: For a random network of 60 nodes in the presence of noise, prediction error E_{nz} as a function of normalized measurements R_m, after excluding neighbors of the hidden node. Inset shows, for $R_m = 0.5$ as the arrow indicates, the variances of the coefficient vectors for all the nodes. There is only one hidden node in all cases and its neighborhoods are node No. 4, No. 14 and No. 30, which correspond to the tall bars. Uniform noise of amplitude 1% is added to the payoff vector and the measurement matrix.

We now address the effects of noise. As shown in Fig. 4.3, for a network of 60 nodes, the prediction errors decrease with the amount of the measurement data, with relative error of about 10% in the weights of the existing links. In this case, the links for all nodes except the neighborhoods of the hidden node are still predictable. The variances of the predicted vectors, as shown in the inset of Fig. 4.3, are larger compared with those in noiseless situation, but the neighborhoods of hidden node still have significantly larger variances than the others, indicating that the hidden

node can still be detected reliably when the noise amplitude is weak as compared with the coupling strength of the hidden node. It is also possible to distinguish the effects due to noise and hidden node. The idea is that, when a hidden node is present, its influences on other nodes in the network are distinct, while the effect of noise on different nodes is statistically uniform and independent.

While we have demonstrated the principle of detecting hidden nodes using the setting of evolutionary-game dynamics, our formulation is general and applies to other types of network dynamics. For example, we have applied our method to detecting hidden node in networks with continuous-time oscillatory nodal dynamics by expanding $\tilde{\mathbf{g}}_i$ and $\mathbf{g}_{ij}$ into power series and obtaining a similar matrix $\mathbf{G}$, where the system response is the derivatives of the corresponding variables [29, 30]. The unknown coefficients vectors $\mathbf{a}$ can then be solved, giving rise to full knowledge about the nodal and coupling dynamics. By examining the variances in $\mathbf{a}$, we can confirm and locate precisely the location of the hidden node in the network. We have applied our method to both continuous- and discrete-time oscillatory dynamics. Extensive numerical tests indicate that the method is robust with respect to different complex-network structures such as random, scale-free and clustered topologies, and large variations in the network size as well.

4.4 Conclusion

In summary, we have developed a completely data-driven approach to detecting hidden nodes in complex networks, which are inaccessible to external observation or measurement. The basic idea is to locate the immediate neighbors of the hidden node through reconstruction of the dynamical processes on these nodes that generate the time series or data. Because of their direct links with the hidden node, information used for the reconstruction is incomplete, leading to anomalies and instabilities in the

prediction of their dynamics, which can then be used to infer that they are in the immediate neighborhood of the hidden node. Our reconstruction process is based on compressive sensing. Detecting hidden or black-boxed objects is an extremely challenging but fascinating task in science, and our work opens an avenue to addressing this problem in complex network science and engineering.

Chapter 5

INFERRING HIDDEN NODES IN COMPLEX NETWORK IN THE PRESENCE OF NOISE

5.1 Hidden Node Detection in Noisy Environment

When dealing with an unknown complex system that has a large number of components organized hierarchically and interacting with each other, curiosity demands that we ask the following question: are there hidden objects that are not accessible from the external world? The problem of inferring the existence of hidden objects from observations is quite challenging but it has significant applications in many disciplines of science and engineering. Here by "hidden" we mean that no direct observation of or information about the object is available, and so it appears to the outside world as a black box. However, due to the interactions between the hidden object and other components in the system which are observable, it may be possible to utilize "indirect" information to infer the existence of the hidden object and to locate its position with respect to objects that can be observed. The difficulty to develop effective solutions is compounded by the fact that the indirect information on which any method of detecting hidden objects relies can be subtle and sensitive to changes in the system or in the environment. In particular, in realistic situations noise and random disturbances are present. It is conceivable that the "indirect" information can mix with that due to noise or be severely contaminated. The presence of noise thus poses a serious challenge to detecting hidden nodes, and some effective "noise-mitigation" method must be developed.

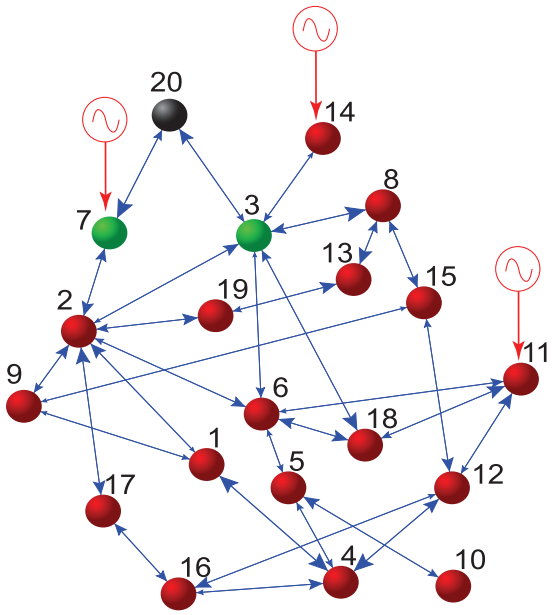

Figure 5.1: An example of a complex network with a hidden node. Time series from all nodes except hidden node #20 can be measured, which can be detected when its immediate neighbors, nodes #3 and #7 are unambiguously identified. Nodes #7, #11, and #14 are driven by local noise sources.

To formulate the problem in a concrete way and to gain insights into the development of a general methodology, we note that the basic principle underlying the detection of hidden objects is that their existence typically leads to "anomalies" in the information that can be directly accessed from the system. Simultaneously, noise, especially local random disturbances applied at the nodal level, can also lead to large variance in the directly available information. This is so because, a hidden node is typically connected to a few nodes in the network that are accessible to the external world, and a noise source acting on a particular node in the network may also be regarded as some kind of hidden object. Thus, the key to any detection methodology is to identify and *distinguish* the effects of hidden nodes on observable information from those due to *local* noise sources.

In this project, we focus on complex networks and develop a general method to differentiate hidden nodes from local noise sources. This problem is intimately related to the works on reverse engineering of complex networks, where the goal is to uncover the full topology of the network based on simultaneously measured time series [10, 11, 12, 13, 14, 15, 16, 17, 112, 18, 19, 20, 21, 104, 30, 27, 113]. Our method to distinguish the effects of hidden node and local noise sources is based on the recent work [85] on utilizing compressive sensing [31, 22, 114, 25] to detect hidden nodes in the absence of such noise sources. To explain our method in a concrete setting, we use the network configuration shown schematically in Fig. 5.1, where there are 20 nodes, the couplings among the nodes are weighted, and the entire network is in a noisy environment, but a number of nodes also receive relatively strong random driving. We assume an oscillator network so that the nodal dynamics are described by nonlinear differential equations, and that time series can be measured simultaneously from all nodes in the network except one, labeled as #20, which is a hidden node. Detecting the presence and locating the position of the hidden node are equivalent to identifying its immediate neighbors, which are nodes #3 and #7 in Fig. 5.1. Note that, in order to be able to detect the hidden node based on information from its neighboring nodes, the interactions between the hidden node and its neighbors must be directional from the former to the latter or be bidirectional. Otherwise, if the coupling is solely from the neighbors to the hidden node, the dynamics of the neighboring nodes will not be affected by the hidden node and, consequently, time series from the neighboring nodes will contain absolutely no information about the hidden node, which is therefore undetectable. The action of local noise source on a node is naturally directional, i.e., from the source to the node.

Our recent work [85] demonstrated that, when the compressive-sensing paradigm is applied to uncovering the network topology [30], the predicted linkages associated

with nodes #3 and #7 are typically anomalously dense, and this piece of information is basically what is needed to identify them as the neighboring nodes of the hidden node. However, the predicted linkages associated with the nodes driven by local noise sources can exhibit behaviors similar to those due to the hidden nodes, leading to significant uncertainty in the detection of the hidden node. To address this critical issue is essential to developing algorithms for real-world applications, which is the aim of this paper. Our main idea is to exploit the principle of *differential signal* and explore the behavior of the predicted weights as a function of the amount of measurement.Due to the advantage of compressive sensing, the required amount of measurement can be quite small and, hence, even our method requires systematic increase of the data amount, it will still be reasonably small. We shall argue and demonstrate that, when the ratio of the predicted weights, which is essentially a kind of differential signal, is examined, those associated with the hidden nodes and nodes under strong local noise will show characteristically distinct behaviors, rendering unambiguous identification of the neighboring nodes of the hidden node.

5.2 Methods

5.2.1 Compressive-sensing Based Method to Uncover Network Dynamics and Topology.

We consider the typical setting of a complex network of N coupled oscillators in a noisy environment. The dynamics of each individual node, when it is isolated from other nodes, can be described as $\dot{\mathbf{x}}_i = \mathbf{F}_i(\mathbf{x}_i) + \xi\eta_i$, where $\mathbf{x}_i \in \mathbb{R}^m$ is the vector of state variables, and η_i are an m-dimensional vector whose entries are independent Gaussian random variables of zero means and unit variances, and ξ denotes the noise

amplitude. A weighted network can thus be described by the following equation:

$$\dot{\mathbf{x}}_i = \mathbf{F}_i(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij} \cdot [\mathbf{H}(\mathbf{x}_j) - \mathbf{H}(\mathbf{x}_i)] + \xi\eta_i, \tag{5.1}$$

where $\mathbf{W}_{ij} \in \mathbb{R}^{m\times m}$ is the coupling matrix between node i and node j, and $\mathbf{H}$ is the coupling function. Defining

$$\mathbf{F}'_i(\mathbf{x}_i) \equiv \mathbf{F}_i(\mathbf{x}_i) - \mathbf{H}(\mathbf{x}_i) \cdot \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij},$$

we have

$$\dot{\mathbf{x}}_i = \mathbf{F}'_i(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij}\mathbf{H}(\mathbf{x}_j) + \xi\eta_i, \tag{5.2}$$

i.e., we have grouped all terms directly associated with node i into $\mathbf{F}'_i(\mathbf{x}_i)$. We can then expand $\mathbf{F}'(\mathbf{x}_i)$ into the following form:

$$\mathbf{F}'_i(\mathbf{x}_i) = \sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i), \tag{5.3}$$

where $\tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i)$ are a set of orthogonal and complete base functions, which are chosen such that the coefficients $\tilde{\mathbf{a}}_i^{(\gamma)}$ are sparse. While the coupling function $\mathbf{H}(\mathbf{x}_i)$ can be expanded in a similar manner, for simplicity we assume that they are linear: $\mathbf{H}(\mathbf{x}_i) = \mathbf{x}_i$. We then have

$$\dot{\mathbf{x}}_i = \sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i) + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij} \cdot \mathbf{x}_j + \xi\eta_i, \tag{5.4}$$

where all the coefficients $\tilde{\mathbf{a}}_i^{(\gamma)}$ and $\mathbf{W}_{ij}$ need to be determined from time series $\mathbf{x}_i$. In particular, the coefficient vector $\tilde{\mathbf{a}}_i^{(\gamma)}$ determines the nodal dynamics and the weighted matrices $\mathbf{W}_{ij}$'s give the full topology and coupling strength of the entire network.

Suppose we have simultaneous measurements of all state variables $\mathbf{x}_i(t)$ and $\mathbf{x}_i(t+\delta t)$ at M different t values at interval Δt apart, where $\delta t \ll \Delta t$, so that the derivative vector $\dot{\mathbf{x}}_i$ can be estimated at each time instant. Equation (5.4) for all the M time

instants can then be written in a matrix form with the following measurement matrix:

$$\mathbf{G}_i = \begin{pmatrix} \tilde{\mathbf{g}}_i(t_1) & \mathbf{x_1}(t_1) & \cdots & \mathbf{x}_k(t_1) & \cdots & \mathbf{x}_N(t_1) \\ \tilde{\mathbf{g}}_i(t_2) & \mathbf{x_1}(t_2) & \cdots & \mathbf{x}_k(t_2) & \cdots & \mathbf{x}_N(t_2) \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ \tilde{\mathbf{g}}_i(t_M) & \mathbf{x_1}(t_M) & \cdots & \mathbf{x}_k(t_M) & \cdots & \mathbf{x}_N(t_M) \end{pmatrix}, \tag{5.5}$$

where the index k in $\mathbf{x}_k(t)$ runs from 1 to N, $k \neq i$, and each row of the matrix is determined by the available time series at one instant of time. The derivatives at different time can be written in a vector form as $\mathbf{X}_i = [\dot{\mathbf{x}}_i(t_1), \cdots, \dot{\mathbf{x}}_i(t_M)]^T$, and the coefficients from the functional expansion and the weights associated with all links in the network, which are to be determined, can be combined concisely into a vector $\mathbf{a}_i$, as follows:

$$\mathbf{a}_i = [\tilde{\mathbf{a}}_i, \mathbf{W}_{1i}, \cdots, \mathbf{W}_{i-1,i}, \mathbf{W}_{i+1,i}, \cdots, \mathbf{W}_{N,i}]^T. \tag{5.6}$$

where $[\cdot]^T$ denotes the transpose. For properly chosen expansion base and a general complex network whose connections are typically sparse, the vector $\mathbf{a}_i$ to be determined is sparse as well. Finally, Eq. (5.4) can be written as

$$\mathbf{X}_i = \mathbf{G}_i \cdot \mathbf{a}_i + \xi\eta_i. \tag{5.7}$$

In the absence of noise or if the noise amplitude is negligibly small, Eq. (5.7) represents a linear equation but the dimension of the unknown coefficient vector $\mathbf{a}_i$ can be much larger than that of $\mathbf{X}_i$, and the measurement matrix will have many more columns than rows. .

5.2.2 Recovering Signal from Noisy Measurement with Compressive Sensing Algorithm.

Conventional wisdom suggests then that the previous linear equation is ill defined. However, since $\mathbf{a}_i$ is sparse, insofar as its number of non-zero coefficients is smaller

than the dimension of $\mathbf{X}_i$, the vector $\mathbf{a}_i$ can be uniquely and efficiently determined by the compressive-sensing paradigm [31, 22, 114, 25].

In this linear equation $\mathbf{X} = \mathbf{G} \cdot \mathbf{a} + \xi$, the stable recovery of the P-dimension sparse vector $\mathbf{a}$ is achievable, according to [22]. Here vector $\mathbf{X} \in \mathbb{R}^{Q\times 1}$ and matrix $\mathbf{G} \in \mathbb{R}^{Q\times P}$ are given, but $P \ll Q$. ξ is a Q-dimension random variable and satisfies Gaussian distribution of zero mean and variance as σ. If the unknown vector $\mathbf{a}$ is sufficiently sparse, we can reconstruct it by solving the following l_1 regularization problem:

$$\min \|\mathbf{a}\|_{l_1}, \text{ subject to } \|\mathbf{G} \cdot \mathbf{a} - \mathbf{X}\|_{l_2} \leq \epsilon, \tag{5.8}$$

where the l_1 norm for a vector $\mathbf{x}$ is defined as $\|\mathbf{x}\|_{l_1} = \sum_{i=1}^{n} |x_i|$, and its l_2 norm is $\|\mathbf{x}\|_{l_2} = \sum_{i=1}^{n} |x_i^2|$. ϵ is the size of the error term ξ. The reconstructed vector $\bar{\mathbf{a}}$ is proved to be within the noise level as $\|\bar{\mathbf{a}} - \mathbf{a}\| \leq C \cdot \epsilon$, and C is a constant.

5.2.3 Detection of Hidden Node.

To motivate our consideration, we note that, a meaningful solution of Eq. (5.7) based on compressive sensing requires the derivative vector $\mathbf{X}_i$ and the measurement matrix $\mathbf{G}_i$ be entirely known which, in turn, requires time series from all nodes. In this case, we say that information required for reconstruction of the complex networked system is *complete*. In the presence of a hidden node, for its immediate neighbors, i.e., the nodes that are directly connected to it, the available information will not be complete in the sense that some entries of the vector $\mathbf{X}_i$ and the matrix $\mathbf{G}_i$ become now unknown. Let h denote the hidden node. For any neighboring node of h, the vector $\mathbf{X}_i$ and the matrix $\mathbf{G}_i$ in Eq. (5.7) now contain unknown entries at the locations corresponding to the index h. For any other node not in the immediate neighborhood of h, Eq. (5.7) is unaffected. When compressive-sensing algorithm is used to solve Eq. (5.7), there will then be large errors in the solution of the coefficient vector $\mathbf{a}_i$

associated the neighboring nodes of h, regardless of the amount of data used. In general, the so-obtained coefficient vector $\mathbf{a}_i$ will not appear sparse. Instead, most of its entries will not be zero, a manifestation of which is that the node would appear to have links with almost every other node in the network. In contrast, for nodes not in the neighborhood of h, the corresponding errors will be small and can be reduced by increasing the data amount, and the corresponding coefficient vector will be sparse. It is this observation which makes identification of the neighboring nodes of the hidden node possible in the noiseless or weak-noise situations [85].

To appreciate the need and the importance to distinguish the effects of hidden node from these of noise, we can separate the term associated with h in Eq. (5.4) from those with other accessible nodes in the network. Letting l denote a node in the immediate neighborhood of the hidden node h, we have

$$\mathbf{X}_l = \mathbf{G}'_l \cdot \mathbf{a}'_l + (\mathbf{W}_{lh} \cdot \mathbf{x}_h + \xi\eta_l), \tag{5.9}$$

where $\mathbf{G}'_l$ is the new measurement matrix that can be constructed from all available time series. While background noise may be weak, the term $\mathbf{W}_{lh} \cdot \mathbf{x}_h$ can in general be large in the sense that it is comparable in magnitude with other similar terms in Eq. (5.4). Thus, when the network is under strong noise, especially for those nodes that are connected to the neighboring nodes of the hidden node, the effects of hidden node on the solution can be mixed up with those due to noise. Also, if the coupling strength from the hidden node is weak, it will be harder to identify the neighboring nodes. For example, hidden node in a network with Gaussian weight distribution will be harder to detect, since the couplings to its neighbors are very likely small thus the variances induced by it will be unsignificant, comparing to those induced by background noise. See **SI** for details.

5.2.4 *Method to Distinguish Hidden Nodes from Local Noise Sources - a Mathematical Formulation.*

For simplicity, we assume that all coupled oscillators share the same coupling scheme and that each oscillator is coupled to any of its neighbors through one component of the state vector only. Thus, each row in the coupling matrix $\mathbf{W}_{ih}$ associated with a link between node i and h has only one non-zero element. Let p denote the component of the hidden node coupled to the first component of node i, the dynamical equation of which can then be written as

$$\begin{aligned}[\dot{\mathbf{x}}_i]_1 = [\sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i)]_1 \quad &+ \quad [\sum_{k\neq i,h}^{N} \mathbf{W}_{ij} \cdot \mathbf{x}_j]_1 \\ &+ \quad w_{ih}^{1p} \cdot [\mathbf{x}_h]_p + \xi\eta_i,\end{aligned} \tag{5.10}$$

where $[\mathbf{x}_h]_p$ is the time series of the pth component of the hidden node, which is unavailable, and w_{1k}^{zx} is the coupling strength between the hidden node and node i. The dynamical equation of the first component of neighbor j of the hidden node has a similar form. Letting

$$\Omega_{ij} = w_{ih}^{1p}/w_{jh}^{1p}, \tag{5.11}$$

be the cancellation ratio, multiplying Ω_{ij} to the equation of node j, and subtracting from it the equation for node i, we obtain

$$\begin{aligned}[\dot{\mathbf{x}}_i]_1 \quad &= \quad \Omega_{ij}[\dot{\mathbf{x}}_j]_1 + \sum_{\gamma} \tilde{\mathbf{a}}_i^{(\gamma)} \cdot \tilde{\mathbf{g}}_i^{(\gamma)}(\mathbf{x}_i) + \sum_{k\neq i,h} w_{ik}^{1p}[\mathbf{x}_k]_p \\ &- \quad \Omega_{ij} \sum_{\gamma} \tilde{\mathbf{a}}_j^{(\gamma)} \cdot \tilde{\mathbf{g}}_j^{(\gamma)}(\mathbf{x}_j) - \Omega_{ij} \sum_{k\neq j,h} w_{jk}^{1p}[\mathbf{x}_k]_p \\ &+ \quad (w_{ih}^{1p} - \Omega_{ij} w_{jh}^{1p}) \cdot [\mathbf{x}_h]_p + \xi\eta_i - \Omega_{ij}\xi\eta_j.\end{aligned} \tag{5.12}$$

We see that terms associate with $[\mathbf{x}_h]_p$ vanish and all deterministic terms on the left-hand side of Eq. (5.12) are known, which can then be solved by the compressive-sensing method. From the coefficient vector estimated from Eq. (5.12), we can iden-

tify the coupling of nodes i and j to other nodes, except for the coupling between themselves since such terms have been absorbed into the nodal dynamics, and the couplings to their common neighborhood are degenerate in Eq. (5.12) and cannot be separated from each other. Effectively, we have combined the two nodes together by introducing the cancellation ratio Ω_{ij}.

To give a concrete example, we consider the situation where each oscillator has three independent dynamical variables, named as x, y and z. For the nodal and coupling dynamics we choose polynomial expansions of order up to n. The x component of the nodal dynamics $[\mathbf{F}'_i(\mathbf{x}_i)]_x$ for node i is:

$$[\mathbf{F}'_i(\mathbf{x}_i)]_x = \sum_{l_x=0}^{n}\sum_{l_y=0}^{n}\sum_{l_z=0}^{n}[a_{l_x l_y l_z}]_x \cdot x_i^{l_x} y_i^{l_y} z_i^{l_z},$$

and the coupling from other node k to the x component can be written as

$$C_{ik}^x = w_{ik}^{xx} \cdot x_k + w_{ik}^{xy} \cdot y_k + w_{ik}^{xz} \cdot z_k,$$

where w_{ik}^{xy} denotes the coupling weight from the y component of node k to the x component of node i, and so on. The nodal dynamical terms in the matrix $\mathbf{G}_i$ are

$$[\tilde{\mathbf{g}}_i]_x = [x_i^0 y_i^0 z_i^0, x_i^1 y_i^0 z_i^0, \cdots, x_i^n y_i^n z_i^n],$$

and the corresponding coefficients are $[a_{l_x l_y l_z}]_x$. The vector of coupling weights is $[\mathbf{W}_{ij}]_x = [w_{ij}^{xx}, w_{ij}^{xy}, w_{ij}^{xz}]$. Equation (5.12) becomes

$$\begin{pmatrix} \dot{x}_i(t_1) \\ \dot{x}_i(t_2) \\ \vdots \\ \dot{x}_i(t_M) \end{pmatrix} \approx \mathbb{G} \cdot \begin{pmatrix} \Omega_{ij} \\ c \\ \tilde{\mathbf{a}}'_i \\ -\Omega_{ij} \cdot \tilde{\mathbf{a}}'_j \\ w_{i1}^{xx} - \Omega_{ij} w_{j1}^{xx} \\ \vdots \\ w_{iN}^{xz} - \Omega_{ij} w_{jN}^{xz} \end{pmatrix}, \tag{5.13}$$

where $\mathbb{G}$ has the following form:

$$\mathbb{G} = \begin{pmatrix} \dot{x}_j(t_1) & 1 & [\tilde{\mathbf{g}}_i(t_1)]_x & [\tilde{\mathbf{g}}_j(t_1)]_x & x_1(t_1) & \cdots & z_N(t_1) \\ \dot{x}_j(t_2) & 1 & [\tilde{\mathbf{g}}_i(t_2)]_x & [\tilde{\mathbf{g}}_j(t_2)]_x & x_1(t_2) & \cdots & z_N(t_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \dot{x}_j(t_M) & 1 & [\tilde{\mathbf{g}}_i(t_M)]_x & [\tilde{\mathbf{g}}_j(t_M)]_x & x_1(t_M) & \cdots & z_N(t_M) \end{pmatrix}, \tag{5.14}$$

and c is the sum of constant terms from the dynamical equations of nodes i and j, and $\tilde{\mathbf{a}}_i'$ is the coefficient vector to be estimated which excludes all constants. Using compressive sensing to solve Eq. (5.13), we can recover the cancellation ratio Ω_{ij} and the nodal dynamic of node i. When Ω_{ij} is known we can then recover the dynamics of node j from the coefficient vector $-\Omega_{ij} \cdot \tilde{\mathbf{a}}_j'$.

We further discuss the extension of this method to system of different nodal dynamics and of multiple hidden nodes, in the **SI**. We show that, our method can manipulate systems of different nodal dynamics, even the derivatives are not defined. For example, evolutional games on complex networks. We show in the **SI** that we can use similar procedure, by replacing the derivatives to the observations, or the agent payoffs, to exam the cancelling factors and further differentiate the hidden nodes from local noise sources. We also show that when some requirements about the couplings between two or more hidden nodes and their neighborhood are satisfied, we can infer the canceling factors for their neighboring nodes, although they are affected by multiple hidden nodes at the same time.

5.3 Results in Coupled Oscillator Networks

We present our results by using coupled oscillator networks. Given such a networked system, we use compressive sensing to uncover all the nodal dynamical systems and coupling functions [30]. This can be done by expanding all the functions into series and calculating, from available time series, all the coefficients in the expan-

sion. The expansion base needs to be chosen properly so that the number of non-zero coefficients is small as compared with the total number N_t of coefficients. All N_t coefficients constitute a *coefficient vector* to be estimated. The amount of data used can be conveniently characterized by R_m, the ratio of actual number M of data points used in the reconstruction, to N_t. See **Methods**.

Our idea to develop an effective procedure to distinguish the effects of hidden node and local noise sources is based on the following observation. Consider two neighboring nodes of the hidden node, labeled as i and j. Because the hidden node is a common neighbor of nodes i and j, the couplings from the hidden node should be approximately proportional to each other, with the proportional constant determined by the ratio of their link weights with the hidden node. When the dynamical equations of nodes i and j are properly normalized, the terms due to the hidden node tend to cancel each other, leaving the normalization constant as a single unknown parameter that can be estimated subsequently. We name this parameter *cancellation ratio* and denote it by Ω_{ij}. As the data amount is increased, Ω_{ij} tends to its true value. Practically we then expect to observe systematic changes in the estimated value of the ratio as data used in the compressive-sensing algorithm is increased from some small to relatively large amount. If only local noise sources are present, the ratio should show no systematic change with the data amount. Thus the distinct behaviors of Ω_{ij} as the amount of data is increased provides a way to distinguish the hidden node from noise and, at the same time, to ascertain the existence of the hidden node. A mathematical formulation of this general principle can be found in **Methods**.

We test our method to differentiate hidden nodes and noise using random networks of nonlinear/chaotic oscillators. To be concrete, we choose the nodal dynamics to be that of the Rössler oscillator, one of the classical models in nonlinear dynamics [115],

which is given by

$$[\dot{x}_i, \dot{y}_i, \dot{z}_i] = [-y_i - z_i, x_i + ay_i, b + z_i(x_i - c)],$$

where $a = 0.2$, $b = 0.2$, and $c = 5.7$ so that the system exhibits a chaotic attractor. The size of the underlying network varies from 20 to 100, and the probability of connection between any two nodes is 0.04. The network link weights are equally distributed in $[0.1, 0.5]$ (arbitrary). Similar results are also achievable on systems of different network sizes, weight distributions and topologies, as discussed in **SI**. Background noise, denoted as ξ is applied (independently) to every oscillator in the network, whose amplitude is from 10^{-4} to 5×10^{-3}, which is weaker in magnitude than the average coupling strength. The tolerance of compressive sensing ϵ is adjusted in accord with the noise strength. Time series are generated by using the standard Heun's algorithm to integrate stochastic differential equations. To approximate the velocity field, we use third-order polynomial expansions in the compressive-sensing formulation.

5.3.1 Detecting Hidden Node from Time Series.

As a concrete example, we consider the network of coupled Rössler oscillators in Fig. 5.1, where background noise is present. Linear coupling between any pair of connected nodes is from the z-component to the x-component. From the available time series (nodes #1-19), we can solve the coefficient vector using some standard compressive-sensing algorithm [116]. In particular, for node i, the terms associated with couplings from the z-components of the other nodes appear in the ith row of the coupling matrix. As shown in Fig. 5.2(a), when the amount of data is $R_m = 0.7$, the network's coupling matrix can be predicted. The predicted links and the associated weights are sparse for all nodes except nodes #3 and #7, the neighbors of the hidden

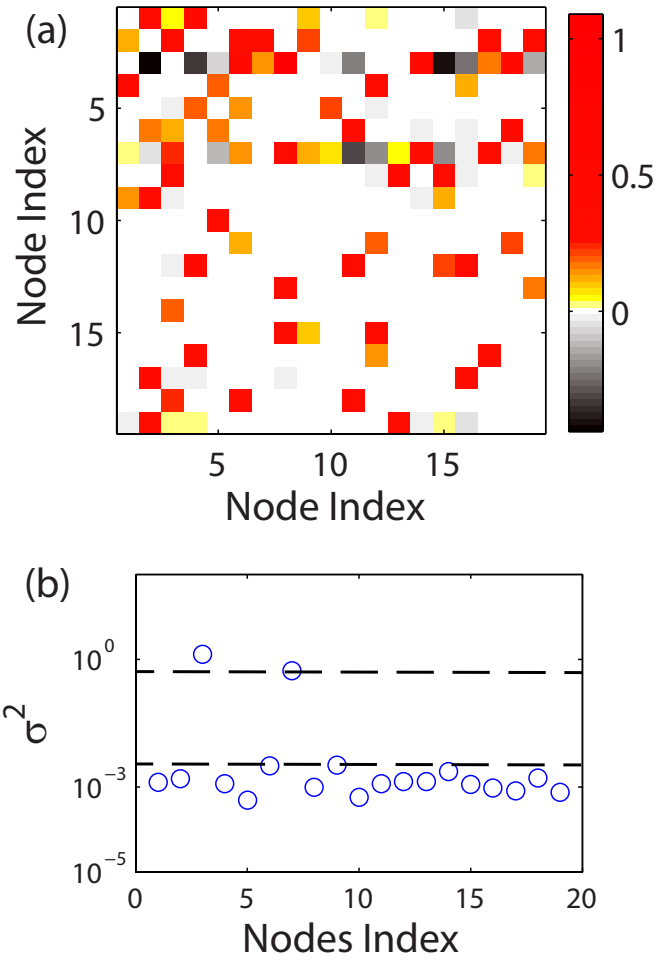

Figure 5.2: For the network in Fig. (5.1), (a) predicted coupling matrix for all nodes except node #20. Time series from nodes #1 to #19 are available, while node #20 is hidden. The predicted weights are indicated by color coding and the amount of data used is $R_m = 0.7$. The abnormally dense patterns in the 3rd and 7th rows suggest that nodes #3 and #7 are the immediate neighbors of the hidden node. (b) Variance σ^2 of predicted coefficients of all the accessible nodes, which is calculated using 20 independent reconstructions based on different segments of the data. The variances associated with nodes #3 and #7 are apparently much larger than those of the other nodes, confirming that these are the neighboring nodes of the hidden node. Their gap can be illustrated with two dash lines in panel (b).

node. While there are small errors in the predicted weights due to background noise, the predicted couplings for the two neighbors of the hidden node, which correspond to the 3rd and the 7th row in the coupling matrix, appear to be from almost all other nodes in the network and some coupling strength is even negative. Such anomalies associated with the predicted coupling patterns of the neighboring nodes of the hidden node cannot be removed by increasing the data amount. However, it is precisely these anomalies which hint at a strong possibility that these two "abnormal" nodes are connected with a hidden node.

While abnormally high connectivity predicted for a node is likely indication that it belongs to the neighborhood of the hidden node, in complex networks there are hub nodes with abnormally large degrees, especially for scale-free networks [5]. In order to distinguish a hidden node's neighboring node from some hub node, we can use the variance of the predicted coupling constants, which can be calculated from different segments of the available data sets. Due to the intrinsically low-data requirement associated with compressive sensing, the calculation of the variance is feasible because any reasonable time series can be broken into a number of segments, and prediction can be made with respect to each data segment. For nodes not in the neighborhood of the hidden node, we expect the variances to be small as the predicted results hardly change when different segments of the time series are used. However, for the neighboring nodes of the hidden node, due to lack of complete information needed to construct the measurement matrix, the variances can be much larger. Figure 5.2(b) shows the variances σ^2 in the predicted coupling strength for all 19 accessible nodes. We observe that the variances for the neighboring nodes of the hidden node, nodes #3 and #7, are all above as the upper dash line and significantly larger than those associated with all other nodes, which are all below the lower dash line. This indicates strongly that these are indeed the neighboring nodes of the hidden node. The gap

between these two dash lines represents the significance of the effects brought by hidden nodes. The larger the gap is, the more unequivocal it is to distinguish the neighbors of hidden nodes from the normal nodes. The results in Fig. 5.2 thus indicate that the possible location for hidden node(s) can be easily identified even in the presence of weak background noise. The significance of hidden node will vary among different systems. The variances introduced by hidden nodes are mainly determined by the coupling strength from them, as it is shown in Fig.S? in SI. It is less relevant to system sizes and coupling topologies as they are shown in SI.

The hidden node significance is also depended on the successful reconstructions of all nodes that are not in the neighborhood of hidden nodes, which determine the lower dash line. To quantify the reliability of the reconstruction results, we investigate how the prediction error in the link weights of all accessible nodes except the predicted neighbors of the hidden node changes with respect to data amount. For an existent link, we use the normalized absolute error E_{nz}, the error in the estimated weight with respect to the true one, normalized by the value of the true link weight. Figure 5.3 shows the results for $N = 100$. All links take weights uniformly from $[0.1, 0.5]$, and the background noise is $\xi = 10^{-3}$. The tolerance for compressive sensing is $\epsilon = 0.5$, which is the optimal when noise strength $\xi = 10^{-3}$. Details about how to determine the optimal tolerance for system of different noise strength be found in **S.I.** and Fig. S?. We see that for $R_m > 0.4$, E_{nz} decreases to the small value of about 0.01, which is set by background noise. As R_m is increased, the error and the variance exhibit similar values, indicating that the reconstructed results are stable. Although the value of E_{nz} does not decrease further toward zero due to noise, the prediction results are reliable in the sense that the predicted weights and the real values match with each other, as shown in the inset of Fig. 5.3, a comparison of the exact weights and the predicted results for all existent links, illustrated by the x-axis and y-axis of dots in

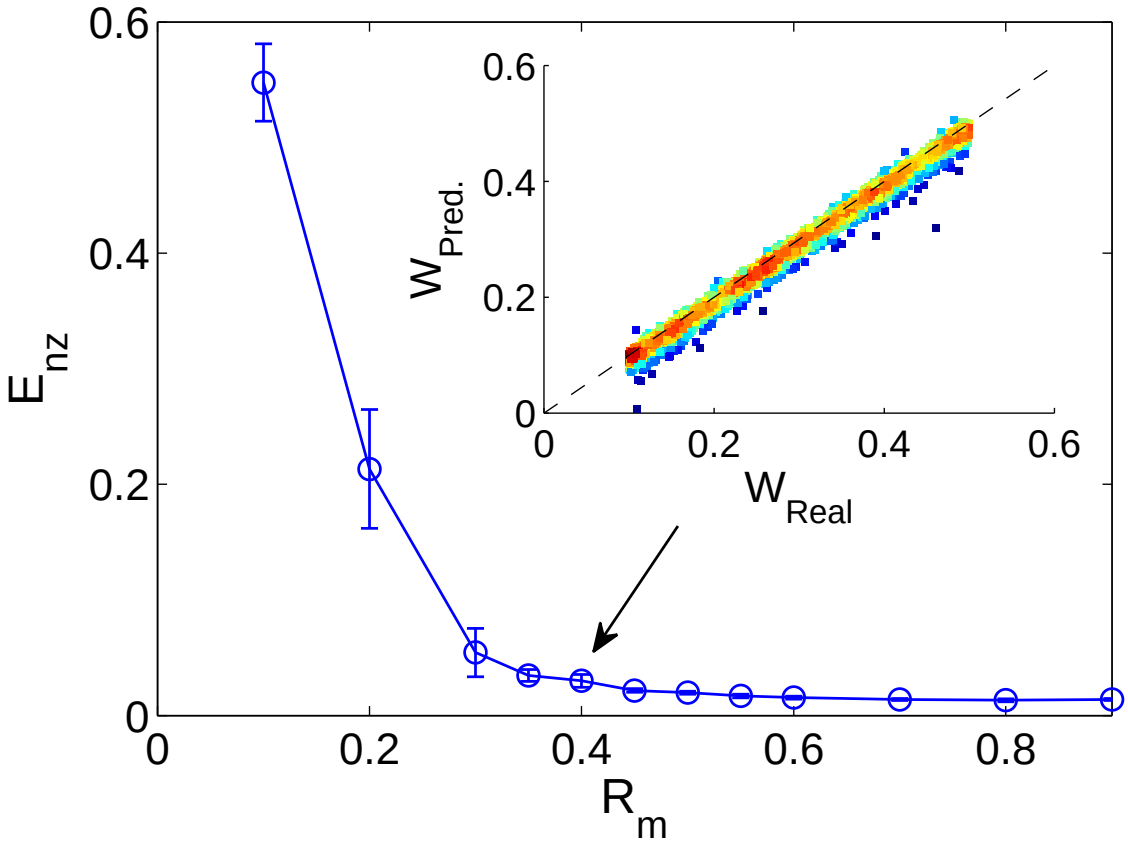

Figure 5.3: For random networks of 100 nodes and uniform weight distribution in $[0.1, 0.5]$, prediction error E_{nz} associated with nonzero coefficients of dynamical equations of all nodes except for the neighboring nodes of the hidden node, as a function of normalized data measurements R_m. The background noise strength is $\xi = 10^{-3}$ for all nodes. All data points are obtained from 10 independent simulations. Inset is a comparison of real and predicted weights for all existent links. Each dot represents one existed link, and its x-axis is the real weight while the y-axis is the corresponding predicted result. The color for each dot is determined by the dot density around it, while the hot color represents high density. The arrow indicates the value of R_m used for calculating the comparisons. The tolerance of compressive sensing algorithm is $\epsilon = 0.5$.

the inset. We see that all the predicted results distributes around the real value, since all dots stay around the diagonal curve $y = x$. The central region of dot distributions have hotter color than the marginal part, which further confirm that the majority of predictions are quite accurate, and only a small portion of predictions fall off the real value. We further show in the **SI** that, stable reconstruction can be achieved on various network sizes, connection topologies and weight distributions, when sufficient measurement data are given.

5.3.2 Differentiating Hidden Node from Local Noise Sources.

When strong noise sources are present at certain nodes, the predicted coupling patterns of the neighboring nodes of these nodes will show anomalies. (Here by "strong" we mean that the amplitudes of the random disturbances are order-of-magnitude larger than that of background noise.) We now demonstrate that our proposed methodology based on the cancellation ratio is effective at distinguishing hidden nodes from local noise sources, insofar as the hidden node has at least two neighboring nodes not subject to such disturbances. To be concrete, we choose a network of $N = 61$ coupled chaotic Rössler oscillators, which has 60 accessible nodes and one hidden node (#61) that is coupled to two neighbors: nodes #14 and #20, as shown schematically in Fig. 5.4. Assume a strong noise source is present at node #54. We find that the reconstructed weights match their true values to high accuracy. We also find that the reconstructed coefficients including the ratio Ω_{ij} are all constant and invariant with respect to different data segments, there is strong indication that the pair of nodes are the neighboring nodes of the same hidden node, thereby confirming its existence.

When there are at least two accessible nodes in the neighborhood of the hidden node which are not subject to strong noisy disturbance, such as nodes #14 and #20,

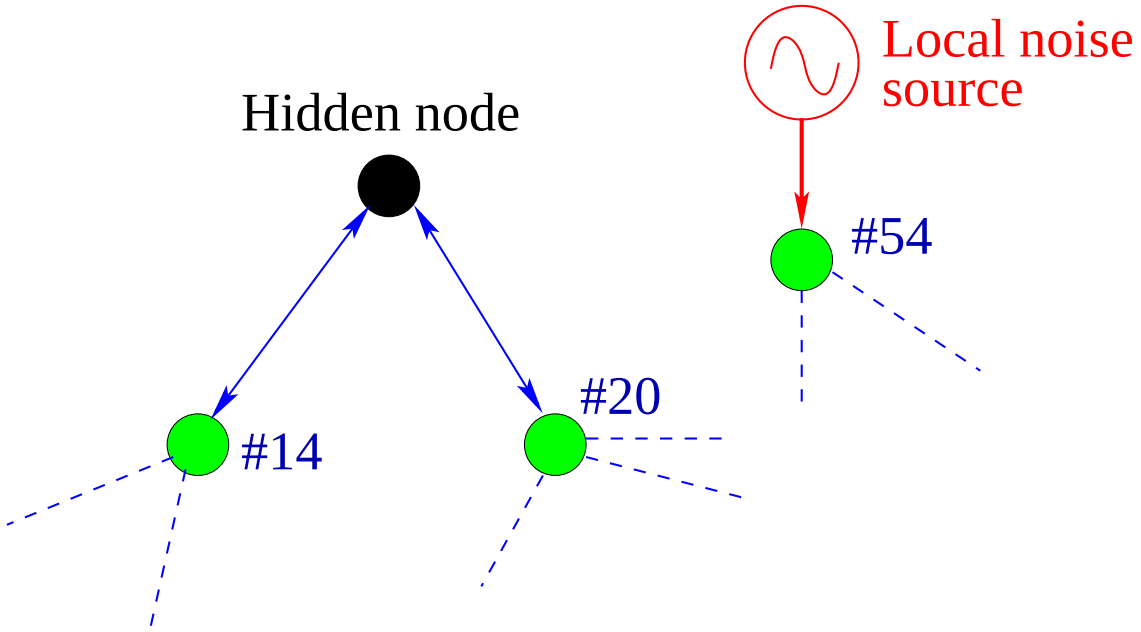

Figure 5.4: Schematic illustration of a hidden node and its coupling configuration with its two neighbors in a random network of $N = 61$ nodes with 60 accessible nodes, which will be used to demonstrate our scheme to distinguish hidden node from local noise sources. A strong noise source is present at node #54.

as the data amount R_m is increased towards 100%, the cancellation ratio should also increase and approach unity. This behavior is shown by the open circles in Fig. 5.5(a). However, when a node is driven by a local noise source, regardless of whether it is in the neighborhood of the hidden node, the cancellation ratio calculated from this node and any other accessible node in the network will show a characteristically different behavior. Consider, for example, nodes #14 and #54. The reconstructed connection patterns of these two nodes both show anomalies, as they appear to be coupled with all other nodes in the network. In contrast to the case where the pair of nodes are influenced by the hidden node only, here the cancellation ratio does not show any appreciable increase as the data amount is increased, as shown by the crosses in Fig. 5.5(a). In addition, the average variance of the predicted coefficient vectors of the two nodes exhibits characteristically different behaviors, depending on whether

any one node in the pair is driven by strong noise or not. In particular, for the node pair #14 and #20, since neither is under strong noise, the average variance will decrease toward zero as R_m approaches unity, as shown in Fig. 5.5(b) (open circles). In contrast, for the node pair #14 and #54, the average variance will increase with R_m, as shown in Fig. 5.5(b) (crosses). This is because, when one node is under strong random driving, the input to the compressive-sensing algorithm will be noisy and its performance will deteriorate. In particular, compressive sensing can perform reliably when the input data are clean, even when they are sparse. Increasing the data amount beyond a threshold is not necessarily helpful, but longer and noisy data sets can degrade significantly the performance. The results in Figs. 5.5(a,b) thus demonstrate that the cancellation ratio between a pair of nodes, in combination with the average variance of the predicted coefficient vectors associated with the two nodes, can effectively distinguish a hidden node from a local noise source. If there are more than one hidden nodes, or cluster of hidden nodes, the procedure to infer the cancelling factors is similar but requiring additional information about their neighboring nodes. The applicability of the cancelling factor method can be extended to systems other than continue oscillatory networks, such as evolutional game systems. See **S.I. for details.**

5.4 Discussion

Our program to differentiate a hidden node from local noise sources and then to infer its existence can be summarized into the following steps:

1. collect time series of all dynamical variables from accessible nodes;

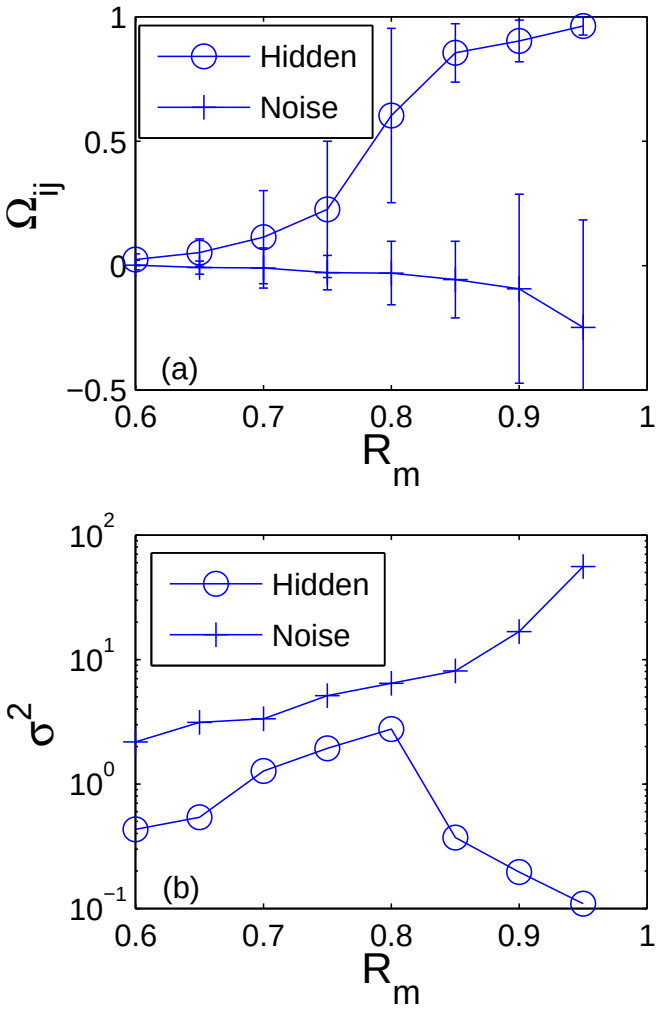

Figure 5.5: For the network described in Fig. 5.4, (a) Predicted values of the cancellation ratio Ω_{ij} obtained from the differential signal of two neighboring nodes of the hidden node (#14 and #20, indicated by circles) and from the differential signal of nodes #14 and #54, where the latter is driven by noise of amplitude $\xi = 10^{-2}$ (crosses). (b) Average variances of the predicted values in the coefficient vectors for the two combinations. The background noise amplitude is $\xi = 10^{-5}$. The results are obtained from 20 independent realizations.

2. hypothesize suitable expansion bases for nodal dynamics and coupling functions taking advantage of physical understanding of the underlying networked dynamical system;

3. construct the measurement matrix and derivative vector from time series, and solve the expansion-coefficient vector using compressive sensing;

4. identify all nodes with abnormally dense connections, and calculate the corresponding variances using independent segments of the available time series to pick out the hub nodes in the network (for those nodes the variances will be much smaller than those of the neighboring nodes of the hidden node or nodes under strong local noise);

5. for all the remaining nodes with abnormally dense connections, calculate the cancellation ratio for all possible node pairs and also the average variance of the predicted coefficient vectors using independent time-series segments for a series of systematically increasing values of the data amount R_m;

6. identify the neighboring nodes of the hidden node as those whose cancellation ratios approach unity and the average variances tend to zero as R_m is increased. For those pairs whose cancellation ratio does not increase and/or the average variances do not decrease with R_m, one node in the pair is under the driving of a local noise source.

Although our method can be applied to diverse network structure, coupling weight distribution, and different dynamical system, there are still a number of limitations with it, as follows.

1. for any accessible node in the network, time series of all dynamical variables are required;

2. reasonable knowledge about the underlying complex dynamics is needed, rendering feasible choosing of a suitable expansion base such that the resulting expansion-coefficient vector is sparse;

3. there must be only a single hidden node or a cluster of hidden nodes with common neighborhoods. If multiple hidden nodes exist, each with its own set of neighboring nodes, our method will work only if there is no overlap between the neighboring sets. Otherwise it will fail.

4. There must be at least two nodes in the neighborhood of the hidden node which are not subject to strong local noise.

Detecting hidden nodes in complex networks whose nodal dynamics, topology, and coupling weights are unknown a priori has vast application potential, such as in social and biological networks. While the limitations discussed above may present a serious obstacle to applying our methodology in realistic situations of complex dynamical networks, inferring the existence of hidden node in the presence of local random perturbations is an extremely challenging problem. Our efforts represent a small step forward in this area of research, where much further work is needed.

Chapter 6

DATA BASED RECONSTRUCTION OF COMPLEX GEOSPATIAL NETWORKS

Complex geospatial networks, networks with components distributed in the real geophysical space, are an important part of the modern infrastructure. Examples include large scale sensor networks and various subnetworks embedded in the Internet. For such a network, often the set of active nodes would depend on time: the network can be regarded as static only in relatively short time scale. For example, in response to certain breaking news event, a communication network within the Internet may emerge, but the network will dissolve itself after the event and its impacts fade away. The connection topologies of such networks are usually unknown but in certain applications it may be desirable to uncover the network topology and to determine the physical locations of various nodes in the network. Suppose time series or signals can be collected from the nodes. Due to the distributed physical locations of the nodes, the signals are time delayed. Is it possible to uncover the network topology, estimate the time delays embedded in the signals from different nodes, and then determine their physical locations? Another issue is the existence of hidden nodes, nodes from which no signals can be collected. Can the existence of a hidden node be ascertained and its location be determined?

Figure 6.1 illustrates a geospatial network. Assume there is a monitoring center that collects data from nodes at various locations, but their precise geospatial coordinates are unknown. The normal nodes are colored in green. There are also hidden nodes that can potentially be the sources of threats (e.g., those represented by dark

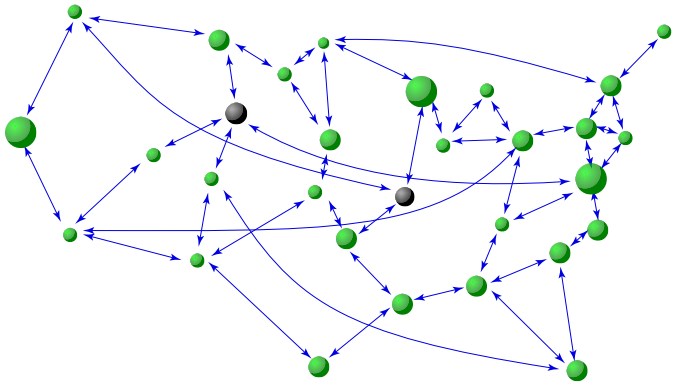

Figure 6.1: A schematic illustration of a complex geospatial network. The connection topology, the positions of the nodes in the physical space, and nodal dynamical equations all are unknown *a priori*, but only time series from the nodes are available. The challenges are to reconstruct the dynamical network, to locate the precise position of each node, and to detect hidden nodes, all based solely on time series with inhomogeneous time delays. The green circles denote "normal" nodes and the dark circles indicate hidden nodes.

circles). The challenging task is to determine the network topology and to locate the hidden nodes, based on time series or data only.

Data based reconstruction of complex networks in general is deemed to be an important problem and has attracted continuous interest, where the goal is to uncover the full topology of the network based on simultaneously measured time series [12, 13, 11, 14, 117, 15, 16, 118, 17, 112, 18, 19, 113, 20, 21, 104, 119, 27, 120, 121, 122, 123, 124, 125]. For instance, methodology was proposed to estimate the network topology controlled by feedback or delayed feedback [118, 119]. Network connectivity can be reconstructed from the collective dynamical trajectories using response dynamics [16, 104]. The approach of random phase resetting was introduced to reconstruct the details of the network structure [20]. For neuronal systems, there was a statistical method to track the structural changes [121, 123]. While many of these previous works required complete or partial information about the dynamical equations of the isolated nodes and their coupling functions, completely data-driven

and model-free methods exist. For example, network structure can be obtained by calculating the causal influences among the time series based on the Granger causality method [117, 124], the transfer entropy method [122], or the method of inner composition alignment [21]. However, such causality based methods are unable to reveal information about the nodal dynamical equations. There were also regression-based methods [126] for systems identification based on, for example, the least-squares approximation through the Kronecker-product representation [82], which would require large amounts of data.

In this project, we develop a methodology based on compressive sensing [25, 22, 114, 24] as a potential solution to estimating time delay and detecting hidden nodes in complex geospatial networks. To be able to fully reconstruct dynamical systems using only time series data is possible because the dynamics of many natural and man-made systems are determined by smooth enough functions that can be approximated by finite series expansions. The task then becomes that of estimating the coefficients in the series representation of the vector field governing the system dynamics. In general, the series can contain high order terms, and the total number of coefficients to be estimated can be quite large. This is in general a very difficult problem to solve. However, if most coefficients are zero (or negligible), the vector constituting all the coefficients will be sparse. The problem of sparse vector estimation can then be solved by the paradigm of compressive sensing [31, 22, 25, 26, 24] that reconstructs a sparse signal from limited observations. Since the observation requirements can be relaxed considerably as compared to those associated with conventional signal reconstruction schemes, compressive sensing has evolved into a powerful technique to obtain high-fidelity signals for applications where sufficient observations are not available.

The compressive sensing paradigm has recently been introduced to the field of network reconstruction for discrete time and continuous time nodal dynamics [29, 30], for evolutionary game dynamics [28], for detecting hidden nodes [85, 87], for predicting and controlling synchronization dynamics [86], and for reconstructing spreading dynamics based on binary data [127]. Differing from these existing works, the focus of the present work is on estimating time delays of the dynamics on various nodes using time series collected from a single location. While there were previous methods of finding time delays in complex dynamical systems, e.g., based on synchronization [128], Bayesian estimation [129], and correlation between noisy signals [120], our compressive sensing based method provides an alternative approach that has the advantages of generality, low data requirement, and applicability to large networks. We demonstrate that our method can yield estimates of the nodal time delays with reasonable accuracy. After the time delays are obtained, the actual geospatial locations of various nodes can be determined by using, e.g., a standard triangular localization method [130]. We expect these results to be useful for applications such as identifying and detecting/anticipating potential geospatial threats [131], an area of importance and broad interest.

6.1 Results

Compressive sensing was developed to solve the following convex optimization problem:

$$\min \|\mathbf{a}\|_1 \quad \text{subject to} \quad \mathbf{G} \cdot \mathbf{a} = \mathbf{X}, \tag{6.1}$$

where $\mathbf{a}$ is a sparse vector to be solved, $\mathbf{G}$ is a (known) random projection matrix, and $\mathbf{X}$ is a measurement vector that can be constructed from the available data, and $\|\mathbf{a}\|_1 = \sum_{i=1}^{N} |\mathbf{a}_i|$ is the L_1 norm of vector $\mathbf{a}$. Compressive sensing is a paradigm of high-fidelity signal reconstruction using only sparse data [31, 22, 25, 26, 24], which

was originally developed to solve the problem of transmitting massive data sets, such as those collected from large-scale sensor arrays. In particular, because of the high dimensionality, direct transmission of such data sets would require broad bandwidth. However, there are common situations where the data sets are sparse. For example, say a data set of N points is represented by an $N \times 1$ vector, $\mathbf{a}$, where N is a large integer. Since $\mathbf{a}$ is sparse, most of its entries are zero and only a small number of k entries are non-zero, where $k \ll N$. One can use a random matrix $\mathbf{G}$ of dimension $M \times N$ to obtain an $M \times 1$ vector $\mathbf{X}$: $\mathbf{X} = \mathbf{G} \cdot \mathbf{a}$, where $M \sim k$. Because the dimension of $\mathbf{X}$ is much smaller than that of the original vector $\mathbf{a}$, transmitting $\mathbf{X}$ would require much smaller bandwidth, provided that $\mathbf{a}$ can be reconstructed at the receiver end of the communication channel.

For our problem of reconstructing complex geospatial networks with time delay, the task is to formulate the problem into the standard compressive sensing form Eq. (6.1). This can indeed be done, e.g., for oscillator networks with weighted interactions and inhomogeneous time delays. After obtaining the time delays, a standard triangular localization method [130] can be employed to locate a large portion of nodes in the network, given that the locations of a small subset of nodes are known. A hidden node can also be detected. All the details can be found in **Methods**.

6.1.1 Reconstruction of Geospatial Networks Based on Compressive Sensing

To be concrete, we present results for continuous-time oscillator networks with time delayed couplings [132, 133], where for every link, the amount of delay is proportional to the physical distance of this link. Consider a link l_{ij} connecting nodes i and j. The weight and time delay associated with this link are denoted as w_{ij} and τ_{ij}, respectively. For a modern geospatial network, the speed of signal propagation is that of light in a proper medium (e.g., optical fiber). The time delay can thus be assumed

to be small and we can use the Taylor expansion to express the delay coupling terms in the networked dynamical system to the first order, e.g., $x_i(t-\tau_{ji}) \approx x_i(t) - \tau_{ji}\dot{x}_i$, where $\dot{x}_i$ is the time derivative. In a suitable mathematical basis, the nodal dynamical equations, the coupling and time delayed terms all can be expanded into series, and the our goal is to estimate all the expansion coefficients. In our formulation of the compressive sensing framework (see **Methods**), the coefficients associated with the nodal dynamical equations, the network link-weights, and time delays are contained in the vector $\mathbf{A} = \{A_i\}$, $\mathbf{B} = \{B_i\}$, and $\mathbf{C} = \{C_i\}$, respectively. The amount of data required depends on the system size and the order of the series expansions, which can be small as compared with the dimension of the coefficient vectors for properly chosen mathematical base.

After obtaining the time delays, we proceed to determine the actual positions of all nodes. If time series are collected simultaneously from all nodes, the estimated coupling delay τ_{ij} associated with the link l_{ij} is proportional to the physical distance $d_{ij} = d_{ji}$ of the link. However, in reality strictly synchronous data collection is not possible. For example, if the signals are collected, e.g., at a location s outside the network with varying time delays τ_{si}, the estimated delays associated with various links in the network are no longer proportional to the actual distances. As we explain in **Methods**, the varying delays due to asynchronous data collection can be canceled and the distances can still be estimated as $d_{ij} = (c/2)(\tau_{ij} + \tau_{ji})$, where τ_{ij} is the signal delay associated with node j from the reconstruction of node i, vice versa for τ_{ji}, and c is the signal propagation speed.

When the mutual distances between nodes have been estimated, we proceed to determine the actual locations of the nodes, e.g., by using the standard triangular localization algorithm [130]. This method requires that the positions of N_B reference nodes be known, the so-called beacon nodes. Starting from the beacon nodes, the

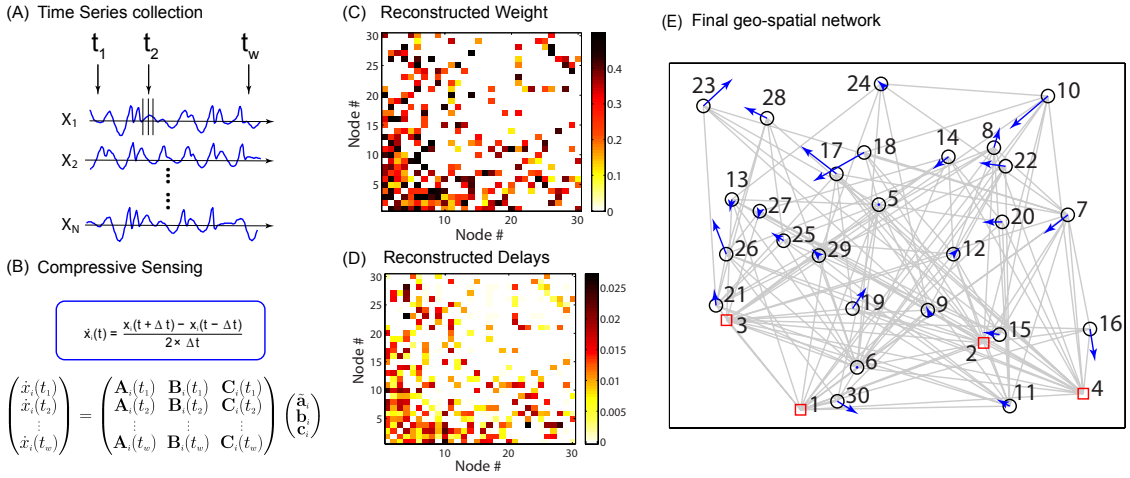

Figure 6.2: Illustration of our method to reconstruct a complex geospatial network from time series. (a) For any node, time series of its dynamical variables are collected at w different instants of time. (b) The corresponding derivatives are approximated using the standard first order Gaussian method, which are needed in constructing the compressive sensing equations. (c,d) An example of link weights and time delays obtained from the reconstructed coefficient vectors, respectively. (e) Given the positions of four beacon nodes (marked as red rectangles), the locations of the remaining nodes (marked as black circles) are determined by using a standard triangularization method. The blue arrows indicate the estimation errors, which point from the actual to the estimated positions. The various coupling terms are illustrated using gray lines. There are in total 30 nodes in the network, connecting with each other via the scale free topology. The average outgoing degree is five. The amount of data used is $R_m = 0.5$.

triangulation algorithm can locate all nodes that are connected to the beacon set with more than three links. The beacon node set can then be expanded with the newly located nodes. The process continues until the locations of all nodes have been determined, or no new nodes can be located. See **Methods**.

Our numerical experiments are set up, as follows. We assume all nodes are distributed in a two dimensional square of unit length. The network topology can be either scale free [5] or random [105], and the network size is varied. For proof of principle, we consider coupled nonlinear oscillator networks by placing, at each node, a nonlinear oscillator, e.g., the Rösseler oscillator, mathematically described by the follow-

ing set of three first-order differential equations: $[\dot{x}, \dot{y}, \dot{z}] = [yz, x{+}0.2y, 0.2x{+}z(x0.2)]$. The coupling weights are asymmetric and uniformly distributed in the interval [0.1, 0.5]. We assign a small threshold to the estimated weight as $w_0 = 0.05$ (somewhat arbitrary), where if the estimated weight is larger (smaller) than w_0, the corresponding link is regarded as existent (nonexistent). We have $d_{ij} = c \cdot \tau_{ij} = c \cdot \tau_{ji}$ and choose c to be 100 (arbitrarily). Without loss of generality, we choose the coupling functions to be linear and for a pair of connected nodes, the interaction occurs between the z-variable of one node and the x-variable of another. The time series used to reconstruct the whole network system are acquired by integrating the coupled delayed differential-equation system [134] with step size 5×10^{-5}. The vector fields of the nodal dynamics are expanded into a power series of order $l_x + l_y + l_z \leq 3$. The derivatives required for the compressive sensing formulation are approximated from time series by the standard first order Gaussian method. To quantify the data requirement, we define R_m as the ratio of the number of data points used to the total number of unknown coefficients to be estimated. The beacon nodes are chosen to be those having the largest degrees in the network, and their positions are assumed to be known.

Figure 6.2 summarizes the major steps required for reconstructing a complex geospatial network using compressive sensing. For illustrative purpose, we use a network of $N = 30$ nodes that are connected with each other in a scale free manner. Oscillatory time series are collected from each node, from which compressive sensing equations can be obtained, as shown in Figs. 6.2(A) and 6.2(B). The reconstructed coefficients for the nodal dynamical equations, as explained in **Methods**, contain the coupling weights $B_{ij} = w_{ij}$ and the delay terms $C_{ij} = -w_{ij} \times \tau_{ij}$. The links with reconstructed weights larger than the threshold w_0 are regarded as actual (existent) links, for which the time delays τ_{ij} can be estimated as $\tau_{ij} = -C_{ij}/w_{ij}$. Repeating this

procedure for all nodes, we can determine the weighted adjacency matrix (that defines the network topology) and the time delay matrix. The estimated adjacency matrix and the time delays are displayed in Figs. 6.2(C) and 6.2(D), respectively, which match well with those of the actual network. We note that the reconstructed time delays are symmetric with respect to the link directions, as shown in Fig. 6.2(D), which is correct as they depend only on the corresponding physical distances. With the estimated time delays, we choose the four largest degree nodes, node #1 $\sim$ #4, as the beacon nodes, so that the locations of all remaining nodes can be determined. The fully reconstructed geospatial network is shown in Fig. 6.2(E), where red rectangles indicate the locations of the beacon nodes. The black circles denote the actual locations of the remaining nodes and the heads of the blue arrows indicate their estimated positions, so shorter arrows mean higher estimation accuracy. The amount of data used is relatively small: $R_m = 0.5$.

6.1.2 Performance Analysis with Respect to Weight and Time Delay Estimates

The performance of our compressive sensing based approach to reconstruction of geospatial networks can be assessed by calculating the errors in the estimated weights and time delays. We define two types of errors: those associated with nonzero terms (existing links, denoted as W_{nz} and D_{nz} for weight and time delay, respectively), and those associated with zero terms (non-existing links, W_z and D_z). In particular, W_{nz} is the error between the estimated and the true weight for an existent link, normalized by the latter, while W_z is the average absolute error associated with the original zero terms in the coefficients. Similar meanings hold for D_{nz} and D_z.

We first study the general behavior of the estimation errors with respect to varying data amount, R_m. Representative results are shown in Fig. 6.3 for $R_m = 0.3$ and $R_m = 0.5$, where panels (A) and (B) are for errors in the weight and time delay

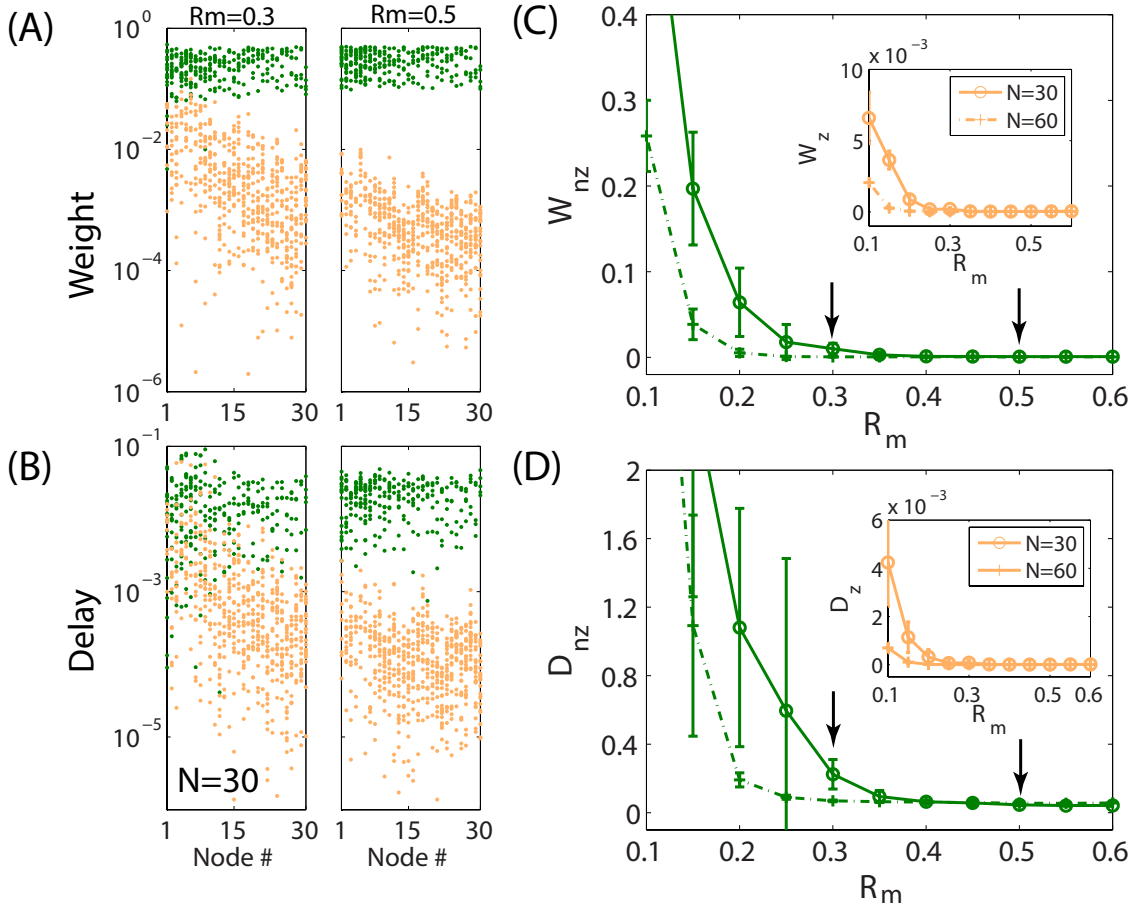

Figure 6.3: Error analysis of network reconstruction and delay time estimation. (A) Predicted incoming coupling strengths for all nodes for $R_m = 0.3$ and $R_m = 0.5$ on a logarithmic scale. The green and orange dots represent the weights of existent and non-existent links, respectively. (B) Predicted coefficients for the nonzero delay coupling terms C_{ij}, marked as green dots, in comparison with the estimated values for zero terms (marked as orange dots). (C) Errors in the estimate of the coupling weights associated with existent and non-existent links, W_{nz} and W_z (inlet), respectively, versus the data amount R_m. (D) Errors in the estimated time delays, where the terms associated with non-zero delay coefficients C_{ij} are normalized by the corresponding weights B_{ij}. The coefficients associated with zero terms (without normalization) are shown in the inlet. The green and orange curves represent results obtained from networks of size $N = 30$ and $N = 60$, respectively. All errors are obtained by averaging over 20 independent network realizations. The black arrows indicate the R_m value used to calculate the results in panels (A) and (B).

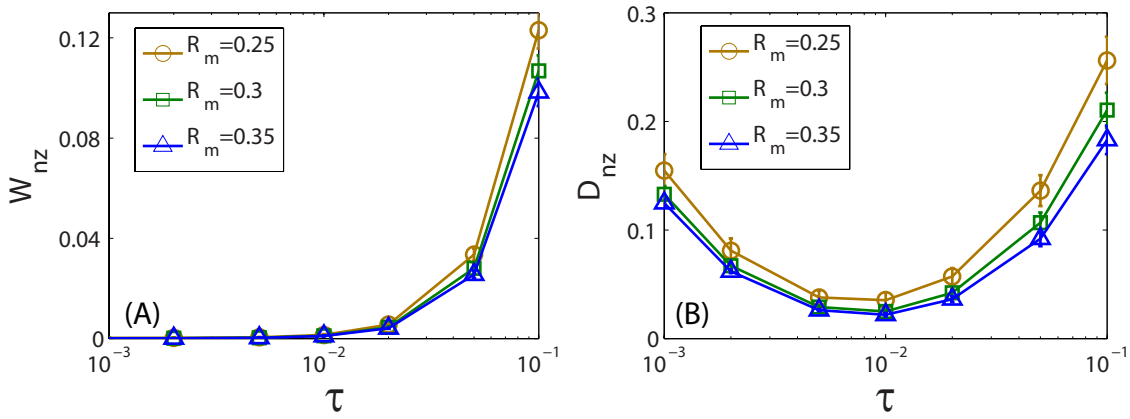

Figure 6.4: Effect of the amount of time delay on reconstruction performance. For networks with uniform time delays, errors of predicted weights (A) and delays (B) versus the length of the actual time delay in the network. Shown in (A) are the normalized errors associated with the nonzero terms in the weights, for several values of R_m. In (B), the errors are associated with the time delays of the existent links. Random networks of size $N = 100$ and connection probability of $P = 0.04$ are used. The results are from three different time series segments, as marked with different symbols. Each data point is the result of averaging over 10 network realizations.

estimation, respectively. For small data amount (left column), the gap between the weights for existent and non-existent links are not well defined, especially for nodes of large degrees. As R_m is increased, the two kinds of weights can be unequivocally distinguished, making possible identification of the existent links (middle column). Ensemble averaged errors in the estimated weights and time delays versus R_m are shown in Figs. 6.3(C) and 6.3(D), respectively. Note that, in Fig. 6.3(D), the terms associated with non-zero coefficients C_{ij} are adjusted by the corresponding weights B_{ij} to compensate the actual coupling delays and the absolute errors associated with the zero coefficients, as shown in the inlet of Fig. 6.3(D), are the averages of the corresponding absolute values of the c_{ij} terms. A general observation is that the various errors decrease rapidly as the data amount is increased, a distinct feature of compressive sensing.

In our mathematical formulation of the compressive sensing based method, the terms containing the time delays are expanded to first order only. The methodology, as it stands now, thus applies to systems with small time delays. To determine an upper bound of the time delay, below which the whole system including various time delays can still be reconstructed faithfully, it is necessary to assess the dependence of the estimation errors on the amount of the time delay. Figures 6.4(A) and 6.4(B) show, for the special case of uniform time delay, errors W_{nz} and D_{nz} versus τ, respectively. We see that the weight errors increase monotonically with τ, especially for $\tau > 10^{-2}$. However, the time delay errors reach minimum for $\tau \approx 10^{-2}$ and begin to increase as τ is increased further. For relatively large time delays, the first-order Taylor expansion becomes less accurate, leading to large errors in the weight and time delay estimation. For small delays, the error in D_{nz} is due to the finite step size used in integrating the delay differential equations.

How does the performance depend on the network size and other characteristics such as the link density? Figure 6.5 shows, for random networks of varying size N and average connection probability P, the errors W_{nz} and D_{nz}. Specifically, for each pair of nodes in the network, their connection probability is given by $p_{ij} \sim p_0/d_{ij}^2$, where d_{ij} is the distance between them, and p_0 is a normalization constant used to fix the average connection probability as P. In this type of "normalized" networks, nodes have a larger tendency to connect to the nearby nodes, as in a real geospatial network. In Figs. 6.5(A) and 6.5(B), the network size varies from $N = 30$ to $N = 100$ while the connection density remains fixed at $P = 0.04$. The errors are illustrated using different colors. When the data amount R_m is increased, the errors decrease rapidly and approach a small constant value when R_m exceeds a certain critical value. We find that optimal reconstruction performance can be achieved for smaller values of R_m for networks of larger size than those of smaller size, indicating that accurate

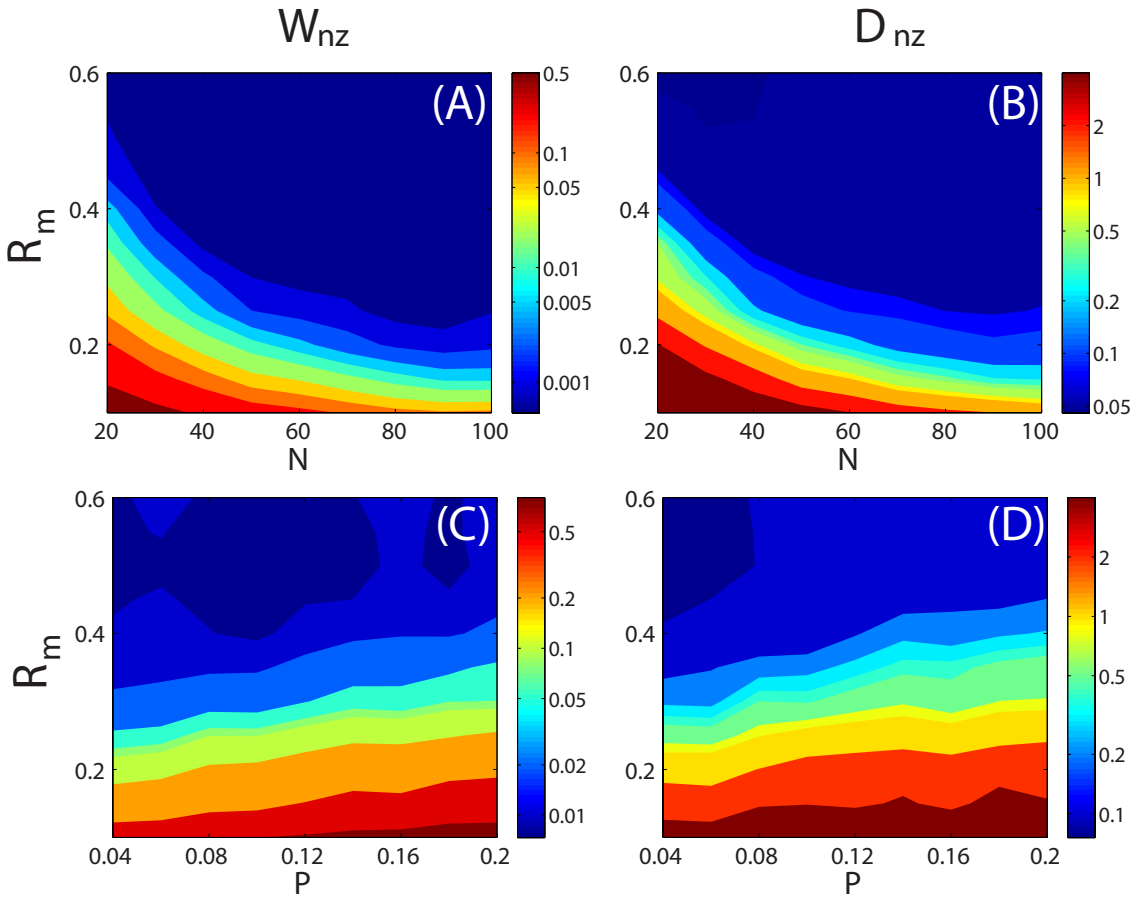

Figure 6.5: Effect of network size on reconstruction performance. (A,B) Errors associated with nonzero terms of weights W_{nz} and time delays D_{nz}, respectively, as the network size N is increased (left to right), for increasing data amount (bottom to top). The networks are random with fixed link probability $P = 0.04$. Cold colors represent small errors. (C,D) Errors in the weights (panel C) and delays (panel D) versus the connection probability P for fixed $N = 30$. All results are obtained by averaging over 20 independent network realizations.

reconstruction of a larger network requires relatively smaller ratio of measurements to the number of unknown coefficients, although the absolute data amounts are larger than those for smaller networks. This is so because, as N is increased, the density of nonzero terms in the dynamical equations and coupling functions decreases for fixed connection density.

As the connection probability P is increased, e.g., from 0.02 to 0.2, for fixed network size (e.g., $N = 30$), larger data amount is required for reasonably accurate reconstruction, as shown in Figs. 6.5(C) and 6.5(D). This is consistent with previous results on reconstruction of complex networks without time delays [29, 30], a feature of the compressive sensing based method.

6.1.3 Error Analysis of Triangulation Algorithm for Nodal Positioning in the Geophysical Space

To locate all nodes in a two-dimensional space requires knowledge of the positions of at least three nodes (minimally four nodes in the three-dimensional space). Due to noise, the required number of beacon nodes will generally be larger. Since node positioning is based on time delays estimated from compressive sensing, which contain errors, the number of required beacon nodes is larger than three even in two dimensions. To quantify the positioning accuracy, we use the normalized error M_r, defined as the medium distance error between the estimated and actual locations for all nodes (except the beacon nodes), normalized by the distributed length L. Figure 6.6 shows M_r versus the fraction R_B of the beacon nodes. The reconstruction parameters are chosen such that the errors in the time delay estimation is $D_{nz} \approx 0.12$. For small values of R_B, the positioning errors are large. Reasonable positioning errors are obtained when R_B exceeds, say, 0.2.

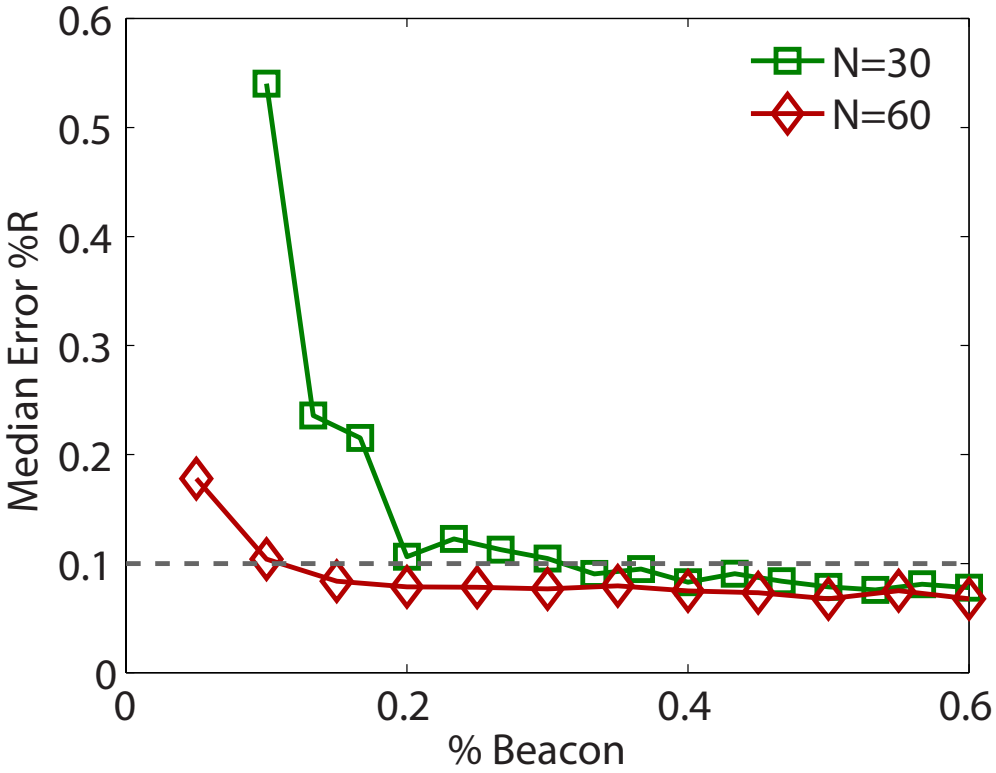

Figure 6.6: Positioning errors. Normalized positioning error M_r, defined as the medium absolute estimated distance error normalized by the distributed length L, as a function of the fraction R_B of the beacon nodes. The networks have the scale free topology with the average outgoing degree $k = 5$. Two values of the network size are used: $N = 30$ and $N = 60$. The beacon nodes are chosen as these having the largest degrees. The time delays are estimated using the data amount $R_m = 0.5$, for which the average error is $D_{nz} \approx 0.12$. The results are obtained by averaging over 10 independent network realizations.

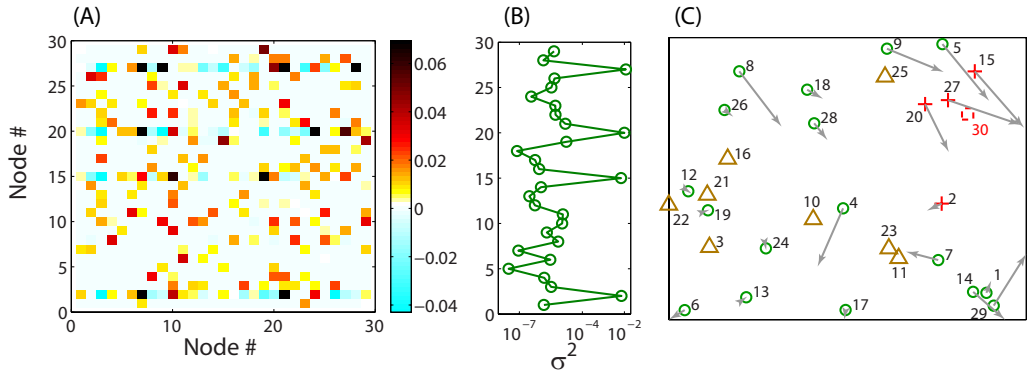

Figure 6.7: Detection of hidden nodes in geospatial networks. For a random network of $N = 30$ nodes, illustration of detecting a hidden node (#30). (A) Reconstructed time delays using time series from 29 externally accessible nodes. (B) Average variance in the reconstructed incoming coupling delays calculated from different segments of the available time series. (C) Estimated positions of all accessible nodes in comparison with the respective actual positions, and the location of the hidden node. The triangles denote the beacon nodes, whose positions are known *a priori*. The green circles denotes "normal" nodes without any hidden node in their immediate neighborhoods, while the cross are direct neighbors of the hidden node. The actual position of the hidden node #30 is marked as a dashed square.

6.1.4 Locating Hidden Node in a Geospatial Network.

To demonstrate the our compressive sensing based approach can be used to ascertain the existence of a hidden node and to estimate its physical location in a geospatial network, we use the model random network in Fig. 6.5. A node is regarded as "hidden" when no time series or other type of direct information can be obtained from it. To detect a hidden node, it is necessary to identify its neighboring nodes [85]. For an externally accessible node, if there is hidden node in its neighborhood, the corresponding entry in the reconstructed adjacency matrix will exhibit an abnormally dense pattern or contain meaningless values. In addition, the estimated coefficients for the dynamical and coupling functions of such an abnormal node will typically exhibit much larger variations when different data segments are used, in comparison with those associated with normal nodes that do not have hidden nodes

in their neighborhood. The mathematical formulation of our method to uncover a hidden node can be found in **Methods**. Initially, there are only 29 time series, one from each of the normal node, and it is not known *a priori* that there would be a hidden node in the network. We proceed to reconstruct the network to obtain the estimated weights and time delays, as shown in Fig. 6.7(A). From the results, we find that the connection patterns of some nodes are relatively dense and the values of weights and time delays are meaningless (e.g., negative values), giving the first clue that these nodes may be the neighboring nodes of some hidden node. To confirm that this is indeed the case, we divide the available time series into a number of segments under the criterion that the data requirement for reconstruction is satisfied for each segment. As shown in Fig. 6.7(B), we observe extraordinarily large variances in the estimated coefficients associated with the abnormal nodes. Combining results from Figs. 6.7(A) and 6.7(B), we can claim with confidence that the four nodes are indeed in the immediate neighborhood of the hidden node, ascertaining its existence in the network.

While the results from Figs. 6.7(A) and 6.7(B) confirm the existence of a hidden node in the network, its geophysical location is still unknown. Note that each of the neighboring nodes of the hidden node is connected to a number of "normal" nodes in the network. We can then use the standard triangularization procedure to determine the locations of all the "abnormal" nodes. Since a geospatial random network has the property that nodes tend to connect with physically nearby nodes, we can deduce that the hidden node must be in the geographical vicinity of the abnormal nodes. In the example in Fig. 6.7, the hidden node (#30, represent by the red square) must then stay near its neighboring nodes (nodes #2, #15, #20 and #27, represented by red crosses), as shown in Fig. 6.7(C), where the normal nodes that are not in the neighborhood of the hidden node are denoted by green circles.

6.2 Methods

6.2.1 Mathematical Framework for Reconstructing Coupled Oscillator Networks with Time Delay.

As proof of principle, we present our reconstruction framework using continuous time oscillator networks. There are N oscillators in the network, and the dynamical process on each node is described by a set of coupled ordinary differential equations (ODEs). (A similar framework can be formulated for other types of dynamical processes, such as evolutionary games [28].) The system can be written as

$$\dot{\mathbf{x}}_i = \mathbf{F}_i[\mathbf{x}_i(t)] + \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij}[\mathbf{x}_j(t-\tau_{ij}) - \mathbf{x}_i(t)], \tag{6.2}$$

for $i = 1, \ldots, N$, where $\mathbf{x}_i \in \mathbb{R}^m$ is the m-dimensional state variable of node i, $\mathbf{F}_i[\mathbf{x}_i(t)]$ is the vector field for its isolated nonlinear nodal dynamics, and the interaction weight between nodes i and j is given by the $m \times m$ weight matrix $\mathbf{W}_{ij} \in \mathbb{R}^{m\times m}$ with its component $w_{ij}^{p,q}$ representing the coupling from the qth component of node j to the pth component of node i. Let the time delay associated with the existent link between nodes i and j be τ_{ij}, regardless of the specific coupling channel between some dynamical variable of node i and some variable of node j. For simplicity, we assume linear coupling functions and causality so that all τ_{ij} $(i, j = 1, \ldots, N)$ are positive. We regroup all terms directly associated with node i into $\mathbf{F}'_i[\mathbf{x}_i(t)]$, where

$$\mathbf{F}'_i[\mathbf{x}_i(t)] \equiv \mathbf{F}_i[\mathbf{x}_i(t)] - \mathbf{x}_i(t) \cdot \sum_{j=1, j\neq i}^{N} \mathbf{W}_{ij}, \tag{6.3}$$

and we expand $\mathbf{F}'_i[\mathbf{x}_i(t)]$ into the following series form:

$$\mathbf{F}'_i[\mathbf{x}_i(t)] = \sum_{\gamma} \tilde{\alpha}^{(\gamma)} \cdot \tilde{\mathbf{g}}^{(\gamma)}[\mathbf{x}_i(t)], \tag{6.4}$$

where $\tilde{\mathbf{g}}^{(\gamma)}[\mathbf{x}_i(t)]$ represents a suitably chosen set of orthogonal and complete base functions such that the coefficients $\tilde{\alpha}^{(\gamma)}$ are sparse. To proceed, we approximate

$\mathbf{x}_j(t-\tau_{ij})$ as

$$\mathbf{x}_j(t-\tau_{ij}) \approx \mathbf{x}_j(t) - \tau_{ij}\dot{\mathbf{x}}_j(t), \tag{6.5}$$

so all the coupling terms with inhomogeneous time delays associated with node i can be written as

$$[\sum_{j=1,j\neq i}^{N} \mathbf{W}_{ij}\mathbf{x}_j(t-\tau_{ij})]_p \equiv \sum_{j=1,j\neq i}^{N} [\mathbf{b}_{ij}\mathbf{x}_j(t) + \mathbf{c}_{ij}\dot{\mathbf{x}}_j(t)], \tag{6.6}$$

where $\mathbf{b}_{ij} = \mathbf{W}_{ij}$ and $\mathbf{c}_{ij} = -\mathbf{W}_{ij}\tau_{ij}$. Equation (6.2) can then be written in the following compact form:

$$\dot{\mathbf{x}}_i(t) = \sum_{\gamma} \tilde{\alpha}^{(\gamma)} \cdot \tilde{\mathbf{g}}^{(\gamma)}[\mathbf{x}_i(t)] + \sum_{j=1,j\neq i}^{N} [\mathbf{b}_{ij}\mathbf{x}_j(t) + \mathbf{c}_{ij}\dot{\mathbf{x}}_j(t)], \tag{6.7}$$

which is a set of linear equations, where $\tilde{\alpha}^{(\gamma)}$, $\mathbf{b}_{ij}$ and $\mathbf{c}_{ij}$ are to be determined. If the unknown coefficient vectors can be reconstructed accurately, we will have complete information about the nodal dynamics as represented by $\mathbf{F}'[\mathbf{x}(t)]$, the topology and interacting weights of the underlying network as represented by $\mathbf{W}_{ij}$, as well as the time delays associated with the nonzero links because of the relations $\mathbf{W}_{ij} = \mathbf{b}_{ij}$ and $\tau_{ij} = -\mathbf{c}_{ij}/\mathbf{b}_{ij}$.

As an illustrative example, we consider the case where each individual nodal dynamical system is three-dimensional with variables x, y and z. For the first component x_i of node i, we have the following series expansion at time t:

$$\begin{aligned}
\dot{x}_i &= (\tilde{a}_i)_{000} \cdot x_i^0 y_i^0 z_i^0 + \cdots + (\tilde{a}_i)_{333} \cdot x_i^3 y_i^3 z_i^3 + (b_{i1})_1 x_1 + (b_{i1})_2 y_1 \\
&+ (b_{i1})_3 z_1 + \cdots + (b_{iN})_1 x_N + (b_{iN})_2 y_N + (b_{iN})_3 z_N + (c_{i1})_1 \dot{x}_1 \\
&+ (c_{i1})_2 \dot{y}_1 + (c_{i1})_3 \dot{z}_1 + \cdots + (c_{iN})_1 \dot{x}_N + (c_{iN})_2 \dot{y}_N + (c_{iN})_3 \dot{z}_N,
\end{aligned}$$

where b_{ii} and c_{ii} are excluded. The formula contains three parts: the power series of isolated nodal dynamics with coefficients $\tilde{a}_i$, terms of all other coupled nodes' variables with coefficients b_{ij}, and terms of derivatives of the coupled nodes as represented by

c_{ij}. Assuming that measurements $x_i(t)$, $y_i(t)$ and $z_i(t)$ at a set of time instants $t_1, t_2, \cdots, t_w$ are available, we write

$$\begin{aligned}\mathbf{A}_i(t) &= [x_i(t)^0 y_i(t)^0 z_i(t)^0, \cdots, x_i(t)^3 y_i(t)^3 z_i(t)^3], \\ \mathbf{B}_i(t) &= [x_1(t), \cdots, x_N(t), y_1(t), \cdots, y_N(t), z_1(t) \cdots, z_N(t)],\end{aligned}$$

and

$$\mathbf{C}_i(t) = [\dot{x}_1(t), \cdots, \dot{x}_N(t), \dot{y}_1(t), \cdots, \dot{y}_N(t), \dot{z}_1(t) \cdots, \dot{z}_N(t)]$$

to obtain a compact expression

$$\mathbf{X} = \mathbf{G} \cdot \mathbf{a}_i + \xi, \tag{6.8}$$

where

$$\mathbf{X} = \begin{pmatrix} \dot{x}_i(t_1) \\ \dot{x}_i(t_2) \\ \vdots \\ \dot{x}_i(t_w) \end{pmatrix} = \begin{pmatrix} \mathbf{A}_i(t_1) & \mathbf{B}_i(t_1) & \mathbf{C}_i(t_1) \\ \mathbf{A}_i(t_2) & \mathbf{B}_i(t_2) & \mathbf{C}_i(t_2) \\ \vdots & \vdots & \vdots \\ \mathbf{A}_i(t_w) & \mathbf{B}_i(t_w) & \mathbf{C}_i(t_w) \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{a}}_i \\ \mathbf{b}_i \\ \mathbf{c}_i \end{pmatrix} + \xi, \tag{6.9}$$

and ξ represents the error introduced by the series approximation.

In general, the connection pattern of a complex network is far sparser than the all-to-all coupling configuration, so typically most elements of $[\tilde{\mathbf{a}}_i, \mathbf{b}_i, \mathbf{c}_i]^T$ are zero. In addition, the error ξ is small and can be regarded as a noise term.

6.2.2 Compressive Sensing Algorithm in Presence of Noise

The compressive sensing algorithm can be used to solve a sparse vector $\mathbf{a}$ from the ill-conditioned linear equation under noise: $\mathbf{X} = \mathbf{G} \cdot \mathbf{a} + \xi$, where ξ is a random process. Reliable recovery of the sparse vector $\mathbf{a}$ can be achieved [31, 22, 25] by solving the following l_1 regularization problem:

$$\min \|\mathbf{a}\|_{l_1}, \quad \text{subject to} \quad \|\mathbf{G} \cdot \mathbf{a} - \mathbf{X}\|_{l_2} \leq \varepsilon, \tag{6.10}$$

where l_1 norm for a vector $\mathbf{x}$ is defined as $\|\mathbf{x}\|_{l_1} = \sum_{i=1}^{n} |x_i|$ (its l_2 norm is $\|\mathbf{x}\|_{l_2} = \sum_{i=1}^{n} |x_i^2|$) and ε is a threshold value determined by the noise amplitude. The reconstructed vector $\bar{\mathbf{a}}$ lies within the range: $\|\bar{\mathbf{a}} - \mathbf{a}\| \leq C \cdot \varepsilon$, where C is a constant.

In order to apply the compressive sensing algorithm to solve Eq. (6.9), the restricted isometric property must be ensured, which can be realized by normalizing each column of the matrix $\mathbf{G}$ by the L_2 norm of that column: $(\mathbf{G}')_{ij} = (\mathbf{G})_{ij}/L_2(j)$ with $L_2(j) = \sqrt{\sum_{i=1}^{M} [(\mathbf{G})_{ij}]^2}$. We thus have $\mathbf{X} = \mathbf{G}' \cdot \mathbf{u}'$ with $\mathbf{u}' = \mathbf{u}L_2$. After $\mathbf{u}'$ is determined through some standard compressive-sensing algorithm, the coefficients $\mathbf{u}$ are given by $\mathbf{u}'/L_2$. Substituting $\mathbf{a}_i$, $\mathbf{b}_{ij}$ and $\mathbf{c}_{ij}$ back into Eq. (6.7), we obtain the the nodal dynamics, coupling weights and the delays associated with the first dynamical variable of node i. For the remaining two variables for this node and all variables for other nodes in the network, a similar procedure can be followed.

6.2.3 Triangle Localization Method

Given the positions of k reference nodes (or beacon nodes) (x_k, y_k), and their distances $d_{i,1}, d_{i,2}, \cdots d_{i,k}$ to the target node i, we can calculate the position of node i using the triangular localization method [130], for k larger than the space dimension. In general, we will need to solve the least squares optimization problem $\mathbf{H} \cdot \mathbf{x}_i = \mathbf{b}$, where $\mathbf{x}_i = [x_i, y_i]^T$ is the position of node i, and $\mathbf{H} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_k]^T$ is the position vector corresponding to the set of beacon nodes, where $\mathbf{b} = 0.5 \times [D_1, D_2, \cdots, D_k]^T$ and $D_k = d_{ik}^2 - y_k^2 + x_k^2$. Here $[\ldots]^T$ means transpose.

To locate the positions of all nodes in the network, we start with a small set of beacon nodes whose actual positions are known and the distances associated with all links in the network. Initially we can locate the nodes that are connected to at least three nodes in the set of beacon nodes, insofar as the three reference nodes are not located on a straight line. When this is done, the newly located nodes can

be added into the set of reference nodes and the neighboring nodes can be located by the new set of beacon nodes. We iterate this process until the positions of all nodes are determined or no more qualified neighboring nodes can be found. For a general network, such initial beacon sets may not be easily found. A special case is scale free networks, for which the initial beacon set can be chosen as these nodes with the largest degrees. For a random network, we can also choose the nodes of the largest degree as the initial beacon node set, and use a larger beacon set to locate most of the nodes in the network. If the network topology is not known, we use the following simple method to select the set of beacon nodes: we estimate the distances from one node to all other unconnected nodes using the weighted shortest distance and then proceed with the triangular localization algorithm. There are alternative localization algorithms based on given distances, e.g, the multidimensional scaling method [135, 136].

6.2.4 Asynchronous Data Collection

In real applications, the requirement to collect time series simultaneously will usually not be met. Consider the typical situation where the data are collected from a fixed external node, denoted by s, which has varying distances to the signal sources. The signals that arrive at the external node at time t were actually sent out by the sources at time $t-\tau_{is}$, where τ_{is} is the varying transmission delay associated with the distances from node i to s. In general, τ_{is} is unknown *a priori* because the location of node i needs to be determined.

In the reconstruction of the dynamical process and connections associated with node i, the time series substituted into Eq. (6.7) are in fact $x_i(t-\tau_{is})$ and $x_j(t-\tau_{js})$, for $j = 1, 2, \cdots, N$ and $j \neq i$. The delay coupling terms can approximated as

$$\mathbf{x}_j(t-\tau_{is}-\tau_{ij}) = \mathbf{x}_j(t-\tau_{js}-\tau_{is}-\tau_{ij}+\tau_{js}) \approx \mathbf{x}_j(t-\tau_{js}) - \tau'_{ij}\dot{\mathbf{x}}_j(t-\tau_{js}),$$

where τ_{ij} is the actual delay, and τ'_{ij} is the estimated delay. From Taylor expansion we have $\tau'_{ij} = \tau_{is} - \tau_{js} + \tau_{ij}$. Similarly, for node j, the estimated delay is $\tau'_{ji} = \tau_{js} - \tau_{is} + \tau_{ji}$. Because $\tau_{ij} = \tau_{ji}$, we can eliminate the effect of τ_{is} and τ_{js} by averaging the two estimated delays, as $\tau_{ij} = \tau_{ji} = (\tau'_{ij} + \tau'_{ji})/2$. After we obtain the time-delay matrix $\{\tau'_{ij}\}$, we can convert its elements into the actual elements so as to obtain the corresponding distances $\{d_{ij}\}$.

6.2.5 Locating a Hidden Node in a Random Geospatial Network

We proceed to reconstruct the network using our compressive sensing framework, treating the system as if there was no hidden node. As demonstrated in Fig. 6.7, the neighboring nodes of the hidden node will exhibit abnormally dense connection patterns. We then use multiple time series segments to calculate the variance in the reconstructed coefficient vectors for all nodes. In particular, the variance σ_i associated with node i is defined as

$$\sigma_i = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}\sum_{k=1}^{N}(w_{ik} - \tilde{w}_{ik})^2},$$

where T is the number of data segments used, N is the network size, $\tilde{w}_{ij}$ is the average weights over T realizations. The variance associated with the time delays can be calculated in a similar way. The neighboring nodes of the hidden node are those with abnormally dense connection patterns *and* significantly larger variances than others.

6.3 Discussion

Given that data are available from a large number of components of a complex networked system which are distributed in the geophysical space, can the network structure be reconstructed, the locations of all nodes be determined, and hidden

nodes be detected and ascertained? We address these related issues by developing a compressive sensing based approach. In particular, assume that time series or signals from nodes in the network are collected at a single location. Our approach enables not only the network topology to be reconstructed, but also the various time delays of the signals from distinct nodes be estimated. A standard triangularization procedure can then be used to determine the locations of the nodes in the geospatial network based on the time delay estimates. We also demonstrate that the existence of a hidden node, from which no signal or time series is externally accessible, can be inferred and ascertained by identifying all nodes in its immediate neighborhood. The location of the hidden node can then be estimated, as nodes in a geospatial network tend to be locally connected.

We stress that, for data based reconstruction of complex geospatial networks, a significant challenge is that the available time series are time delayed, due to the finite speed of physical signals. One unique contribution of the present work, which goes beyond those of previous works on compressive sensing based reconstruction of complex networks [29, 30, 28, 85, 87, 127], is demonstration that inhomogeneous time delays in a complex network can be estimated reliably using compressive sensing. With information about the time delays, one can determine the geophysical locations of the nodes.

Chapter 7

CONTROL CELL FATE DIFFERENTIATION

7.1 Cell Fate Determination in Synthetic Gene Networks

Controlling nonlinear complex systems is an essential scientific problem that attracted much attention. However, most works focus on the linear control scheme which does not apply to nonlinear systems. In this article, we develop a nonlinear control scheme in the light of the attractor network which depicts the hidden geometry of the nonlinear dynamical system from the viewpoint of control utilizing bifurcation processes. We firstly identify a universal set of elementary controls associated with temporary (or short-term) single parameter adjusting, each of which induces the spontaneous transition of the system from one state (attractor) to the other through bifurcation. Then, combining all the possible elementary controls together, we generate an attractor network of the system, in which each node corresponds to an attractor and the weighted directed link between a given pair of attractors represents the elementary control that realizes the transition from the origin to the target. In the light of a clear geometry of attractor network, the *controllability* of the system is vividly visualized. Furthermore, the experimentally realizable control strategy to steer the system among attractors can also be designed accordingly, which just requires temporary control signal upon parameters (rather than the unpractical direct interference to the state of the system). Further interesting issues such as the efficiency or cost of control paths, and the weighted-shortest path can also be evaluated. Examples from paradigm model systems of gene regularity networks, real biosystems and the related experiments all approve the effectiveness of our nonlinear control scheme.

The attractor network framework provides a method to measure the controllability of high-dimensional nonlinear systems, and moreover, supports as a realistic criterion for nonlinear control to biosystem, technical system, or even social systems.

In a demonstration of how a GRN can be used to investigate an intricate network property, here, we use synthetic biology approaches to explore possible mechanisms for stochastic and irreversible cell fate determinations in a multistable system. First we combined experimental characterization and mathematical modeling to calibrate dynamic parameters of three different mutual inhibitory gene networks constructed using our previously developed promoter library approach [39]. With the aim to initialize the cell population on the basin boundary, we utilized the natural regulation machinery for galactose metabolism in yeast to completely shut off the synthesis of all proteins in the synthetic network in glucose-supplemented media, and hence initialize the system close to the basin boundary. The predicted stochastic differentiation was then experimentally verified in all three gene networks by moving the cells, after initial growth to steady state in glucose-supplemented media, to galactose-supplemented media with the TetR-inhibitor anhydrotetracycline (ATc) to ensure bistability.

7.2 Results

7.2.1 Bistable Regions Located by Showing Hysteresis

Each multistable gene network can be viewed as an energy potential landscape with multiple local minima, each representing one specific cellular state, i.e. cell fate [49]. Cells operate on these landscapes and eventually settle into one of the minima, choosing their cell fates until they transition to another state in response to a perturbation, signal, or even inherent noise. Which local minimum a cell settles into depends largely on where the cell starts its growth on this landscape, i.e. its

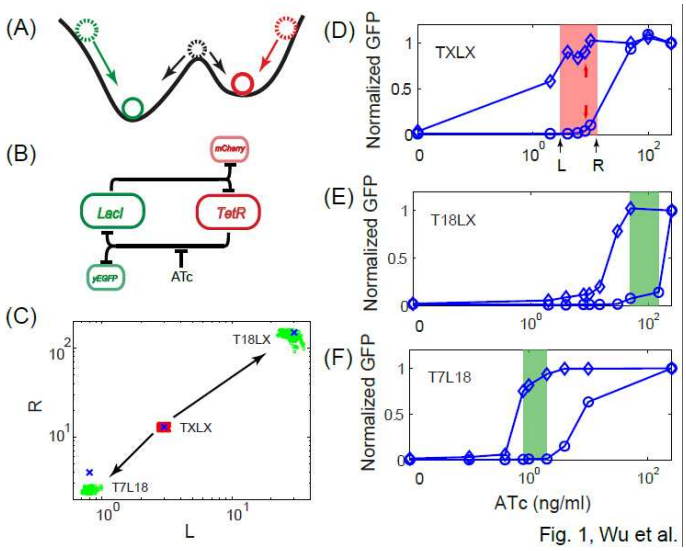

Figure 7.1: Bistable systems experimentally verified by showing hysteresis. A. When fluctuations of movements (inherent noise) are small, initial conditions of free moving marbles (colored dashed circles) on a bistable energy landscape can determine their final steady states (solid circles, analogous to GFP ON and mCherry ON states of our synthetic gene network). However, when the initial conditions are on the tip of barrier between two local minima (black circle), the marble will drop into one of the two local minima randomly (suggested by black arrows), even with minimal amount of noise. B. The schematic diagram of an engineered yeast mutual inhibitory network. LacI and TetR proteins repress each others expression. yEGFP and mCherry are also under control of these promoters and hence indicate the abundance of LacI and TetR proteins. ATc inductions can be used to block TetR (see SI for details).C. Left bound (L) of the bistable region is plotted versus right bound (R) in log scale for all three strains. Blue crosses represent experimentally observed values and colored dots represent model fitted (red) or predicted (green) values. Each dot represents a prediction by one parameter set. Overlap of dots and crossed shows accuracy of predictions. D.E.F. Average green fluorescence of gated cells (pre-treated with 250 ng/ml ATc) at steady states are plotted as diamonds for three different strains. Similarly, data for cultures first treated with no ATc are marked as circles. In all three strains, there is a range where the same dose of ATc induction produces different levels of yEGFP, the sign of hysteresis. Colored shaded regions indicate experimentally observed (D) or model predicted (E, F) bistable region for each strain. In D, Red arrows in point to two data points of TXLX with 8 ng/ml ATc induction, whose full histograms are shown in Fig. 7.2B in purple and green. Choices of L and R are illustrated on the x-axis.

initial conditions. Typically when inherent noise is not too strong, such cell fate determination is fairly predictable and deterministic, depending on initial conditions (Fig. 7.1A, green and red circles). However, when the initial conditions happen to be on the boundary between local minima, the eventual outcome of the cell could become stochastic, regardless of how weak inherent noise is. This is analogous to uncertain marble movements when placed on the top of the barrier between two local minima (Fig. 7.1A, black circle). Such physical intuition has not, however, been observed or realized in either natural or engineered biological systems.

To investigate the possibility of initial condition-dependent random cell fate determination, we choose to use the simplest multistable biological systems, bistable gene networks, to test this hypothesis in cells. Based on our previous work of an engineered yeast GRN with a mutual inhibitory motif [39], three modified versions were constructed with both yeast enhanced green fluorescent protein (yEGFP) and mCherry red fluorescent protein as reporters for LacI and TetR (Fig. 7.1B and S1). To verify bistability and determine the optimal dosage of inducer for balanced steady states, we first tested the gene network TXLX. In this synthetic gene network, the repressors, TetR and LacI, inhibit the expression of each other by binding to their corresponding operator sites, Otet and Olac, placed within engineered GAL1 promoters [39]. The promoters were chosen from our previously generated promoter library [39]. In this library, a GAL1 promoter with Otet operator site (labeled TX) was engineered to form the foundation of 20 variants (labeled T1 to T20). Similarly, a library of LacI controlled promoters (labeled LX and L1 to L20) was also engineered. All of these promoters can be regulated by corresponding repressors and only differ in their maximal expression levels and leakage under repression. The choices of promoter combinations therefore fully determine network characteristics. As readouts, fluorescent protein reporters are under control of TX and LX promoters for all three

strains tested so that they can track LacI and TetR dynamics and also generate strong signals.

Two sets of experiments are designed to demonstrate that the system is capable of hysteresis, an indicator of bistability [36, 41]. Because we previously showed that the networks exhibit the default state of TetR ON (GFP OFF) [39] and IPTG would have no effect at steady state, here we chose ATc inductions as the tuning parameters to probe the systems bistability region. TXLX cultures treated with full ATc induction (250 ng/ml) in galactose-supplemented media for 48 hours were re-diluted into media containing 0, 2, 4, 6, 8, 10, 50, 100, and 250 ng/ml of ATc. Using flow cytometry, measurements of yEGFP were taken after the fluorescence levels become stable in each condition (diamonds in Fig. 7.1D. See SI for experimental details). It can be seen that full ATc induction successfully tilts the balance towards LacI and produced LacI dominant cell cultures (high yEGFP expression). It is also clearly demonstrated that the cultures remain LacI dominant after growth in media with the ATc concentration as low as 4 ng/ml, but fail to maintain the state with 0 and 2 ng/ml ATc induction. Meanwhile, similar experiments were also carried out for cultures treated with no ATc induction for 48 hours. Cells were also re-diluted into media containing various doses of ATc and yEGFP expressions were measured (circles). It can be seen that with no ATc induction, the gene network is TetR dominant (low yEGFP expression). Even with increased doses of ATc induction, cell cultures with up to 10 ng/ml inductions remains low on yEGFP expressions, while only demonstrate increased yEGFP expression with greater than 10 ng/ml ATc inductions. Taken together, it can be seen that, with ATc inductions between approximately 3 and 13 ng/ml, the gene network responds to ATc inductions in an initial condition-dependent fashion, demonstrating hysteresis and hence verifying bistability (see also Fig. S4).

Experimental data were used by the mathematical model to calibrate parameters to accurately locate the bistable region (Blue cross and red dots in Fig. 7.1C. See section II of SI for details about parameter fitting.). This fully quantitative description of our gene networks makes it possible to predict bistable regions for other gene networks. Using calibrated parameters and only adjusting promoter strengths to reflect different constructs, bistable regions are predicted for network T18LX (green dots in Fig. 7.1C and shade in Fig. 7.1E) and T7L18 (green dots in Fig. 7.1C and shade in Fig. 7.1F). To test the accuracy of model predictions, similar experiments were conducted to test hysteresis and locate bistable regions for both constructs. As illustrated in Fig. 7.1E, the construct of T18LX shows a bistable region approximately between 30 and 150 ng/ml ATc, while the construct of T7L18 show a bistable region approximately between 0.6 and 4 ng/ml ATc (Fig. 7.1F). The fact that both experimental measurements are consistent with model predictions demonstrates the predictive power of our model and also builds a solid foundation for further predictions with gene expression stochasticity taken into account.

7.2.2 Model Predicts Ways to Achieve Stochastic and Irreversible Cell Fate Determination

With parameters fitted against experimental data, the model is numerically mapped into a quasi-potential that directs evolution of protein abundances (Fig. 2A). The potential at each point is defined as the trajectory length from this point to its final steady state without stochasticity [137]. Analogous to a marble moving in a landscape in response to gravity, protein abundance changes are directed by the vector field (illustrated as white arrows, see SI and Fig. S7 for details), which is different from gravitational forces as explained in [138]. This map visualizes a more complete picture of a bistable landscape where the dark blue basin bottoms represent two stable steady

states while other regions represent slopes directing the system to the steady states. This saddle-like landscape has one ridge and one valley visualized as white dashed and solid lines, which are known mathematically as the manifolds of this dynamical system [139]. While the latitudes on the solid line form a double-well potential as depicted in Fig. 7.1A, the dashed line is the basin boundary, mathematically termed the separatrix, that divides the landscape into two basins with separate local minima. Cells initialized within each basin should eventually settle into respective basin bottoms. The intersection of the ridge and valley (marked by a black arrow) corresponds to the barrier tip in Fig. 7.1A, mathematically known as the unstable steady states (USS) [139]. As hypothesized above, cells initialized at USS could show random cellular state determination even with minimal level of stochasticity. However, such a specific initial condition is difficult to realize experimentally despite careful parameter calibrations because initialization at this point requires complete and accurate control of protein abundances in a living cell, which is non-trivial to achieve using chemical inductions.

The vector field ensures that trajectories initialized near the ridge will follow it approaching the USS [139]. Therefore we hypothesize that, given some stochasticity, cells initialized on or near the separatrix would first approach the USS along the ridge and then diverge randomly to different cellular states. This eliminates the need to be initialized exactly on the USS for stochastic differentiation. To computationally test this hypothesis, the model was expanded to incorporate gene expression stochasticity using the Gillespie algorithm [140] to simulate the temporal dynamics of cellular state determination. It can be seen in Fig. 7.2A that two isogenic cells (illustrated as two solid lines) starting from the same initial condition (no LacI and no TetR) near the separatrix first follow similar trajectories approaching the unstable steady states and then diverge onto distinct trajectories and finally different cellular states, two separate

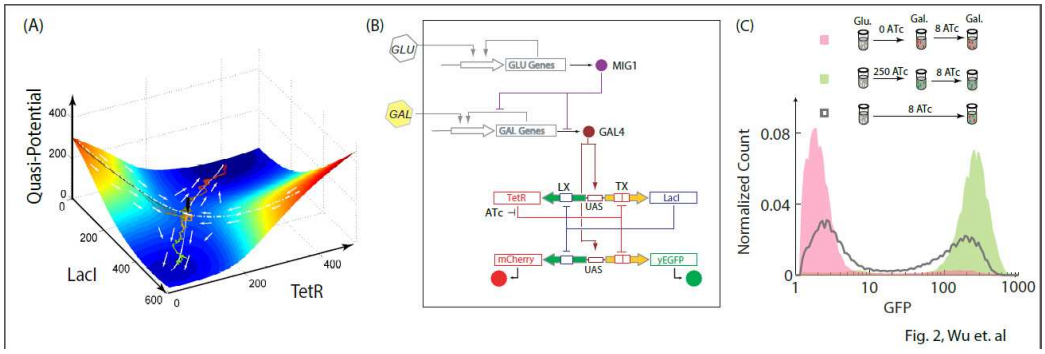

Figure 7.2: Model-predicted stochastic cell fate determination and experimental verifications. A. Based on the ODE model with 8 ng/ml ATc induction, derivatives of state variables are mapped onto an energy quasi-potential landscape that directs the systems evolution. Altitude is color-mapped with cold color indicating lower energy potential. The dashed and solid white lines illustrate the location of the ridge and valley in the landscape. The black arrow indicates the location of unstable steady state. Trajectories of two identical stochastic simulations from the same (0,0) initial conditions are superimposed onto this landscape as solid lines, which are also color-coded to match corresponding single cell fluorescence signal as the cells LacI and TetR concentrations evolve along the landscape. White arrows illustrate the vector field. Landscapes with different doses of inductions are included in Fig. S9. B. Simplified schematics of complete inhibition of GAL1 and UAS regulated genes by glucose culturing. C. After initial growth in glucose-supplemented media to reach log phase, cultures were grown in galactose-supplemented media without ATc induction (pink) and with 250 ng/ml ATc (green) for 48 hours before being moved into galactose-supplemented media with 8 ng/ml ATc until steady state. These two cultures showed different levels of fluorescence, suggesting hysteresis, but similar homogeneous response with a unimodel distribution. Cells moved into the same final condition directly from glucose-supplemented media, however, showed bimodal distribution (gray curve), suggesting an initially uniform population diverged into two distinct populations with either low or high fluorescence outputs, consistent with stochastic simulation predictions.

local minima colored in dark blue. The coloring of the solid lines representing changing color of detectable fluorescence reporter signals. After reaching steady state, one cell will emit a strong red fluorescence signal and the other cell will emit a strong green fluorescence signal. This result computationally verified the hypothesis that as long as initial conditions are on or near the separatrix, isogenic cells can randomly settle into different cellular states even with low-level stochasticity.

To further study cellular state determination when initial conditions are not in close proximity to the basin boundary, stochastic simulations were carried out with initial conditions further away from the basin boundaries. It is shown that, from these initial conditions, distinct and isogenic cells always settle into the same steady state despite the same model parameters and noise levels (Fig. S8). This prediction is also consistent with experiments in the hysteresis experiments. Flow cytometry data collected in Fig. 7.1D all illustrate uni-modal distributions, indicating all cells homogeneously settled into one state (Fig. S10).

7.2.3 Experimental Validations Exploit Natural Yeast Metabolism Regulatory Mechanisms

The challenge of testing our hypothesis is experimental realization of specific initial conditions on the basin boundary. Imperfect regulation by inducers and some leaky expression of all genes in the network when in galactose-supplemented media make it difficult to realize such initial conditions. Here we choose to utilize the natural yeast glucose-galactose metabolism switch mechanism to help us achieve the specific initial condition of no LacI and no TetR in the cell, which is (0,0) on Fig. 7.2A and, based on the model predictions, resides on the basin-boundary of the landscape.

The promoter library core to our GRN is based on the GAL1 promoter, which has been characterized extensively by others [141] and has the ability of being tightly-inhibited when glucose is present as the only carbon source. In the presence of glucose and absence of galactose, galactose metabolism in yeast is completely turned off by the natural GAL metabolic regulatory network, which coordinates expression and repression of GAL promoters via upstream activating sequences (UAS) (Fig. 7.2B). For our GRN, glucose strongly represses all promoters and results in no expression of any gene within our engineered network. As experimentally illustrated in Fig.

S11, expression of yEGFP is fully-repressed, comparable to blank control, in the presence of glucose as the only carbon source, with significantly reduced detection of fluorescence compared to when repressed in galactose-supplemented media without any ATc. Growing our engineered yeast constructs in glucose-supplemented media therefore essentially places these systems on the (0,0) coordinate of LacI-TetR levels, and this will become a point on the basin boundary when the cells are transitioned into galactose-supplemented media with appropriate ATc induction concentrations.

In addition to galactose in the media, it is also shown in Fig. 7.1D that a specific range of ATc concentrations are needed for our systems to be bistable. Numerical simulations also suggest that the stochastic cell fate determination is the most pronounced, i.e., with the highest possibility of being experimentally realized, when the separatrix divide the whole LacI-TetR space into two basins with roughly equal areas. Such requirements typically can be achieved through tuning of levels of inducers. Therefore, we chose 8 ng/ml as the ATc concentration to test the hypothesis because it is about half way between lower and upper bounds of bistability. TXLX cells initially grown in glucose-supplemented media were washed and directly moved into galactose-supplemented media with 8 ng/ml ATc induction. Flow cytometry measurements were taken after 60 hours growth (Fig. 7.2C, gray curve). It can be seen that isogenic cells from the same initial conditions formed two distinct populations expressing completely different levels of yEGFP, one with low expression and one with high expression. In comparison, cells grown in galactose-supplemented media with and without ATc inductions before being moved into galactose-supplemented media plus 8 ng/ml ATc inductions only have homogeneous expressions of yEGFP (Fig. 7.2C, green and pink histograms), demonstrating the irreversibility of cell fate choices. Prolonged growths for all three strains were also carried out and further verified the permanency of such cell fate decisions (Fig. S12). Despite the same in-

ductions of galactose plus 8 ng/ml ATc for all three samples at the end, cells chose clearly different states depending completely on their initial conditions. Starting from high yEGFP expression, cells will remain in this state; starting from low expression, cells will remain in the low state; starting from the basin boundary, cells will choose high or low expression state stochastically.

To verify that this stochastic cell fate determination is not gene network dependent, similar experiments were carried out for two other versions of the yeast bistable gene network: T7L18 and T18LX. As suggested by hysteresis experiments and model predictions, 1.5 ng/ml and 80 ng/ml were chosen for T7L18 and T18LX, respectively, as ATc dosage to form bistable cell fate landscapes. Starting from (0,0) initial conditions realized by growth in glucose-supplemented media, both networks showed well-pronounced bimodal distribution after growth in galactose with respective ATc dosages (Fig. S12), suggesting stochastic cell fate choices. This verifies that as long as cells are initialized on the basin boundary of a multistable system, stochastic cell fate determination can be robustly demonstrated. In addition, each network is tested with ATc induction outside the bistability region and showed only uni-modal distribution (Fig. S12). This verifies that a proper multistable landscape is also a necessary condition for random cell fate choices. In addition, it can be seen that even with large abundance of proteins, intracellular stochasticity can be amplified to dictate cell fate due to the nature of initial conditions and underlying nonlinear system. This complements the common theory of low molecule abundance caused intracellular stochasticity KEBC:2005.

7.2.4 Temporal Measurements Further Illustrate Unique Dynamics

To further verify that the evolution of fluorescence signals from our GRNs actually follows the trajectories predicted in Fig. 7.2A, both flow cytometry and fluorescence

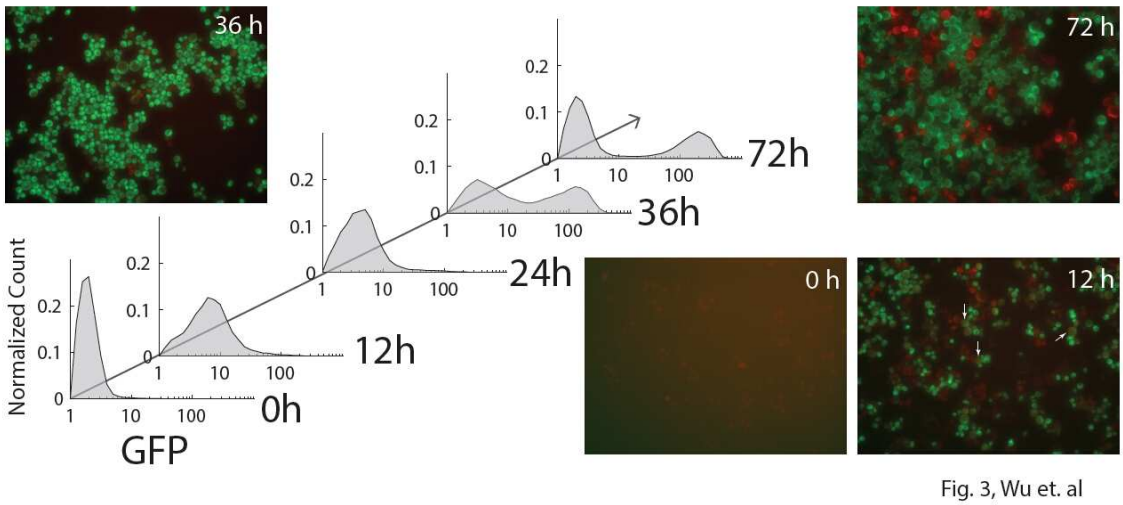

Figure 7.3: Temporal dynamics of random cellular state differentiation demonstrated using both flow cytometry and microscopy imaging. Flow cytometry measurements (histograms) were taken at different time points after TXLX cultures were moved directly into galactose-supplemented media with 8 ng/ml ATc media from glucose-supplemented media. The population gradually increases its fluorescence signal homogeneously until after 36 hours it starts to differentiate into two distinct populations. This observation is consistent with microscopic images taken at these same time points. After 72 hours of growth in galactose-supplemented media with 8 ng/ml induction, cultured cells stably differentiated into two distinct populations, evidenced by bimodal distribution of flow cytometry results and differently colored cells in microscopic images. White arrows in the 12h microscopy image point cells expressing significantly both GFP and mCherry.

microscopy measurements were taken at different time points after each strain was moved into galactose-supplemented media. It can be seen that flow cytometry measurements of TXLX at time 0h have very low yEGFP expression (Fig. 7.3 0h flow cytometry and Fig. S14A black curve). This is consistent with the microscopy image showing only background fluorescence (see SI for imaging details and parameters). This time point represents the initial condition of no LacI nor TetR expression in our cells. After 12 hours of growth in the right media, it can be seen that the whole population of cells show increased yEGFP expression in a homogeneous fashion (Fig. 7.3 12h flow cytometry and Fig. S14A green curve). The microscope image (Fig. 7.3, 12h) also illustrates increased signals for both yEGFP and mCherry. Interestingly,

many cells emit both green and red fluorescence simultaneously at single cell level (labeled by white arrows). These cells show a wide range of fluorescence levels but cant be grouped into distinct populations. This also corroborates with the broad but uni-modal distribution of flow cytometry data at 12 hours. This time point corresponds to the time point in Fig. 7.2A where cells are approaching the unstable steady state, where cells express increased amount of both mCherry and yEGFP and hence result in a homogeneous population. The broad distribution and varied fluorescence levels suggest that cells take different amount of time to travel from the beginning to the USS due to stochasticity. By the time cells spent 36 hours in the media, the pattern of two populations starts to emerge (Fig. 7.3 36h flow cytometry and Fig. S14A blue curve). In addition, one of the peaks shows lower yEGFP expression than 24 hours ago. This clearly demonstrates that the cells start to diverge to different steady states. Cells moving towards LacI dominant state keep increasing their yEGFP expression, while cells moving towards TetR dominant state start to produce more mCherry and inhibit production of yEGFP, eventually making its level to be even smaller than 24 hours ago. This temporal non-monotonic expression of yEGFP by a subpopulation verifies that cells indeed follow a trajectory of approaching the saddle point and then diverging onto two distinct states. Microscopy image is consistent with flow cytometry results, showing cells expressing either yEGFP or mCherry signals. By the time of 72 hours of growth in galactose-supplemented media with 8 ng/ml ATc, the cells clearly formed two populations, illustrated by the two peaks in flow cytometry (Fig. 7.3 72h flow cytometry and Fig. S14A red curve). Correspondingly, microscopy image also shows that cells express either yEGFP or mCherry strongly in a mutually exclusively fashion. The corroboration between flow cytometry and microscopy measurements further supports our predicted temporal dynamics of cells that were started from the (0,0) initial condition on the basin boundary: namely the

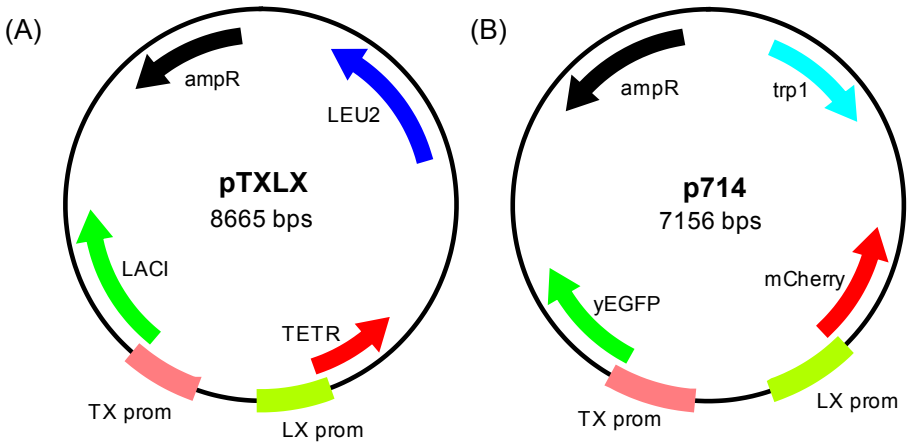

Figure 7.4: Plasmid maps for the switch construct (pTXLX is shown here) and reporter construct (p714). These two plasmids were constructed separately using standard cloning and linearised copies were sequentially integrated into the ura3 locus of the S. cerevisiae YPH500 genome. pT7L18 and pT18LX are identical to pTXLX except for the core promoter sequences of the modified GAL1 promoters..

transient nonmonotonic expression of both fluorescent proteins and eventual strong but random expression of either one.

Temporal flow cytometry measurements were also carried out for T18LX and T7L18. Both show a transient increase and then decrease of yEGFP expression of a subpopulation (Fig. S14B and C), verifying that the unique dynamics is not strain nor network dependent.

7.3 Materials and Methods

7.3.1 Yeast Strains and Plasmid Constructions

Three yeast strains were used in the experiments and contained either the TXLX, T18LX, or T7L18 networks with the 714 red-green reporter construct. All were integrated into S. cerevisiae strain YPH500 (a, ura3-52, lys2-801, ade2-101, trp1D63, his3D200, leu2D1) (Stratagene) with genomic integrations specifically targeted to the

ura3-52 locus [39]. These strains produce yEGFP from the TX promoter, mCherry RFP from the LX promoter (Fig. 7.4). The plasmid construction methods and plasmid maps are described in detail in SI.

7.3.2 Assembly of Gene Networks

The haploid S288C-based Saccharomyces cerevisiae strain YPH500 [142] was used for all experiments. Components of the synthetic gene networks were assembled on pRS-series shuttle vector plasmids [142], using classic restriction enzyme and ligation cloning methods and propagating plasmids in Escherichia coli. All plasmids were integrated into S. cerevisiae at the ura3 genomic locus, growing in synthetic dropout media enabling auxotrophic selection. Colony PCR was used to verify single-copy integration of all components.

Obtaining a yeast strain with a bistable switch controlling green and red fluorescent protein reporters required sequential integration of two plasmids: the switch plasmid (one of pTXLX, pT7L18 or pT18LX) followed by the reporter plasmid (p714). Plasmid maps of these are shown in Supplementary Figure 1 and all have been described previously [39]. The three different yeast strains used in this study, differ only in their integrated switch constructs and these switch constructs themselves only differ in their core promoter regions of the GAL1-based promoters that control TetR and LacI expression. These core promoters have been selected from two previously generated promoter libraries that have been characterised and sequenced and have a wide range of unrepressed and repressed expression levels [39].

7.3.3 yEGFP Induction Experiments

Single yeast colonies for each strain were picked from synthetic dropout selective agar plates containing 2% glucose and were used to inoculate 3 ml of synthetic dropout

media containing 2% glucose (GLU). Colonies were grown at 30 with 300 rpm orbital shaking until reaching an OD600 of 1.0±0.2. A triplicate set of 3 ml cultures synthetic dropout media containing 2% galactose (GAL) and anhydrotetracycline (ATc) at a concentration range of 0 - 250 ng/ml was then prepared and incubated similarly, inoculating from the initial GLU culture to give a starting OD600 of 0.1 - 0.2. All liquid growth media was replaced with fresh media, sugar and ATc every 24 hours.

7.3.4 Flow Cytometry and Data Analysis

Flow cytometry data acquisition was performed with a Becton Dickinson FACScan Analyzer. This machine is equipped for green fluorescence (GFP) measurements. The detector for FSC used E00 channel with SSC at 378 and FL1 at 436 voltages. All data were collected in a log mode. Samples were carried out on a medium flow rate until 100,000 cells had been collected. Data files were analyzed using Matlab with gating. For the bistable region determination experiments, the fluorescence levels have been monitored every 24 h until they become stable. For the stochastic cellular state determinations, samples were taken every12 h for measurement until the fluorescence levels were stable.

For each flow cytometry measurement, we collected data from 100,000 cells. Data were analysed for FL1 fluorescence (measure of GFP level per cell) after first gating a population based on forward scatter (FSC) and side scatter (SSC) in order to reduce extrinsic sources of variation and only focus on cells of similar size, shape, and point in the cell cycle. The gate boundaries used were FSC [650, 800] nm, and SSC [550, 700] nm, marked as a white rectangle in Fig. S2. Flow cytometry data files were analyzed by using Matlab (The MathWorks, Natick, MA). The original log-2 binned FL1 fluorescence intensity values were linearized, and mean values were calculated for each sample.

7.3.5 Model Construction

To model steady state behavior of our engineered networks, we focus on the mutual inhibition between LacI and TetR, and construct a simplified model described by differential equations. The translation and transcription processes of LacI are simplified to be one constant, with the production rate being c_{rl} when the promoter is repressed and c_{il} when induced. Similar assumption is made for TetR, while the rates are c_{rt} and c_{it}, respectively.

To characterize the binding of ATc to TetR and preventing TetR dimer binding to DNA, we use a Hill function to describe the relationship between active ratio of repressor TetR and the inducer concentration ATc: $f_I = (\frac{K_I}{K_I+[ATc]})^m$, where $K_I \equiv k_{ATc}[TetR]$. We assume larger concentration of TetR will need more ATc to deactivate and therefore make K_I in proportion to the concentration of TetR. The probability (or portion of time) of TX promoter not bound by TetR, as a function of the concentration of TetR, can be described as $p_{e,tet} = \frac{k_t^{n_t}}{k_t^{n_t}+[TetR]^{n_t}}$, where k_t represent the TetR concentration needed to make this probability 50%, and n_t describe the nonlinearity of this inhibition. When TX is not bound by TetR, it is fully open and therefore can promote downstream gene expression. Thus under the induction of ATc, the active TetR decreases and probability of open promoter increases. The equation can be expanded to $p_{e,tet} = \frac{k_t^{n_t}}{k_t^{n_t}+([TetR]\cdot f_I)^{n_t}}$, or

$$p_{e,tet} = \frac{k_t^{n_t}}{k_t^{n_t} + \{[TetR] \cdot (1 + \frac{[ATc]k_t}{k_{ATc}[TetR]})^{-m}\}^{n_t}}, \tag{7.1}$$

Similarly, the portion of time that LX is not bounded by LacI is

$$p_{e,lac} = \frac{k_l^{n_l}}{k_l^{n_l} + [LacI]^{n_l}}, \tag{7.2}$$

and k_l represents the LacI concentration needed to make LX bound by LacI 50% of the time, and n_l describes the nonlinearity of this inhibition.

So the production of LacI and TetR can be described by the following two ordinary differential equations:

$$\begin{aligned} \frac{d[LacI]}{dt} &= c_{rl} + p_{e,tet}(c_{il} - c_{rl}) - \delta \cdot [LacI] \\ \frac{d[TetR]}{dt} &= c_{rt} + p_{e,lac}(c_{it} - c_{rt}) - \delta \cdot [TetR], \end{aligned} \tag{7.3}$$

where δ is the degradation rate, which is mainly caused by the volume expansion and thus is approximated as $\delta = 0.002 min^{-1}$. This rate corresponds to a cell doubling time of about 6 hours, which is the doubling time of yeast in galactose media.

7.3.6 Parameter Fitting

For a specific strain we choose, the corresponding synthesis rates, c_{rl}, c_{rt}, c_{il}, c_{it}, can be estimated from experiments by measuring their fully induced and fully repressed gene expression [39]. And the Hill coefficient of induction of ATc, which is the combination of $m \cdot n_t$, can be fitted from the dose response curves [39]. However, the parameters in the binding equilibrium probability $p_{e,lac}$ and $p_{e,tet}$, including n_t, n_l, k_t, k_l and k_{ATc}, need to be fitted from their bistable regions.

Here, we first use randomly generated parameter combinations within a given range to calculate bistable region of TXLX. Specifically, the parameter regions are $n_t \in [1, 5]$, $n_l \in [1, 5]$, $k_t \in [1, 400]$, $k_l \in [1, 400]$ and $k_{ATc} \in [0.01, 1]$, to make sure they have biologically reasonable values. Parameter sets were generated uniformly within the preassigned region and only the ones produce experimentally demonstrated bistability regions within 10% relative error were kept. 200 filtered parameter combinations for TXLX are presented in Fig. 7.5, where the diagonal plots are histograms of all 200 parameters while the others are scatter plot between any two parameters to demonstrate their correlations. We can observe that n_t tends to be slightly bigger than n_l. It reflects the inherently different induction dynamics for TetR and LacI with their

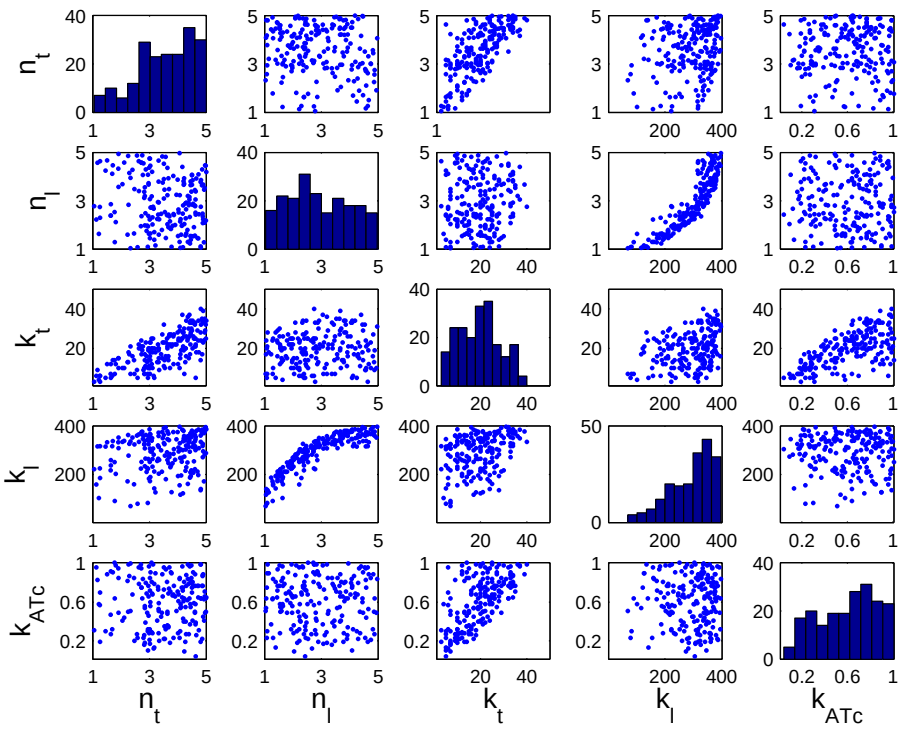

Figure 7.5: Histograms and paired scatter plot for those five estimated parameters. We show 200 filtered parameter combinations for TXLX that generate experimentally demonstrated bistable regions. The candidate parameter combinations are randomly generated within a given region and are kept only if they produce a bistable region that has 10% or smaller relative errors to the experimental ones, which is [3,13].

repestive inducers. Also, k_l tends to take larger values than k_t, which are below 40 as they are shown in the histogram. We aslo observe clear correlation between n_l and k_l, and the one between n_t and k_t.

Using one randomly chosen combination in Fig. 7.5, for example, $n_t = 1.56$, $n_l = 3.35$, $k_t = 11$, $k_l = 264$ and $k_{ATc} = 0.94$, we can plot the bifurcation curve, or the steady state outputs, of LacI concentrations under different ATc concentrations and verify its bistable region. As it is shown in Fig. 7.6, the red curve is the bifurcation curve for TXLX. The solid lines correspond to the set of stable steady states, while the dashed line corresponds to the set of unstable ones. We can see that between [3,13] ng/ml ATc inductions, one ATc concentration corresponds to two stable steady

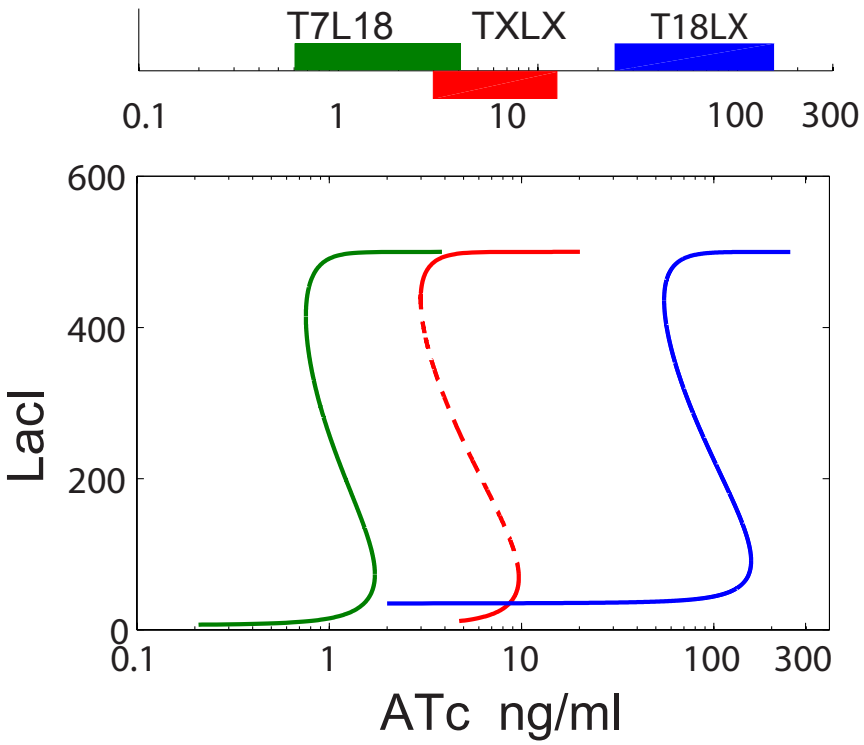

Figure 7.6: Bifurcation curves using filtered (TXLX) or estimated (T7L18 and T18LX) parameters. The parameters for TXLX are $n_t = 1.56$, $n_l = 3.35$, $k_t = 11$, $k_l = 264$ and $k_{ATc} = 0.94$. These curves are generated using the software XPP-AUTO, with standard steps and calculation parameters.

states and one unstable one, a sign of bistability. This ATc region is defined as *bistable region*. If the ATc concentration is outside of this region, there would be only one stable steady output for a specific ATc concentration.

Since the only difference between the three studied strains is the promoter, it is reasonable to estimate the parameters for the other two strains from those in TXLX. Based on the experimental results in [39], where all the strains have the same normalized does response curve, we assume that all three strains have the same nonlinearities on mutual binding and ATc induction, or the same quantities in all Hill functions. However, the concentrations of LacI and TetR are different due to varied promoter strength. So in our model, the parameters k_t, k_l and k_{ATc} need to be

adjusted to normalize the concentration of LacI and TetR to the same values. First we normalize the maximum LacI output. By dividing $c_{il} - c_{rl}$ to both side of the system, we have:

$$\begin{aligned} \frac{d[LacI]'}{dt} &= c'_{rl} + p_{e,tet} - \delta \cdot [LacI]' \\ \frac{d[TetR]'}{dt} &= c'_{rt} + r \cdot p_{e,lac} - \delta \cdot [TetR]', \end{aligned} \tag{7.4}$$

where $c'_{rl} = \frac{c_{rl}}{c_{il}-c_{rl}}$, $c'_{rt} = \frac{c_{rt}}{c_{il}-c_{rl}}$ and $r = \frac{c_{it}-c_{rt}}{c_{il}-c_{rl}}$. Now all the stable maximum LacI output for different strains have the same normalized value, $[LacI]_{max} = 1/\delta$ (since $c_{rl}/(c_{il} - c_{rl}) \ll 1$), thus all strains can share the same k_l. The maximum output of TetR is $[TetR]_{max} = r/\delta$, so we use the following transformations

$$k'_t = \frac{r}{r_0} k_t, \text{ and } k'_{ATc} = \frac{r}{r_0} k_{ATc}, \tag{7.5}$$

to replace the original parameter k_t and k_{ATc}, and guarantee the TetR level in the Hill functions are the same. Here r_0 is from the original strain TXLX used to filter parameters.

With fitted parameters for TXLX and adjusted model to account for strain differences, we can predict bistable regions for the other two strains, T18LX and T7L18, without further parameter fitting. Using the same parameter combination as for TXLX, we model the other two strains with different r according to Eq. 7.5. Then we can draw the bifurcation curves for them and predict their bistable regions. The bifurcation diagrams for the other two strains are shown in Fig. 7.6, and their bistable regions are indicated on the top axis. These predictios are consistent with experimental verifications shown in Fig. 7.1C. Similar to strain TXLX, when the ATc concentration is inside the bistable region, the strain will have two stable steady state outputs.

To directly compare model predictions with the experimental hysteresis curves, we numerically simulated the hysteresis experiments using our model with parameters

mentioned above. We assume that the bistable region locates between ATc= 0 ng/ml and 250 ng/ml for TXLX. Thus when ATc= 0ng/ml, the system will stay in low LacI concentration, no matter what initial condition is, while with ATc=250 ng/ml the system will have stable high LacI concentration. First we simulate the system with 0 ng/ml ATc until steady state, then change ATc concentrations to various levels and simulate the system to its new steady state. This is essentially simulating yeast cells moved into various doses of ATc induction after initial growth in galactose with 0 ng/ml ATc. This produced the dashed curve in Fig. 7.7 (A). The solid curve in Fig. 7.7 (A) is generated similarly by starting the simulations from the stable steady state corresponding to 250 ng/ml ATc and then change ATc inductions to various doses. It corresponds to first grow the yeast cells in galactose with 250 ng/ml, and then move them to galactose with different ATc concentrations. From the curves we can identify the bistable regions, which are ATc region that corresponds to two stable steady LacI concentrations. Similar procedures are repeated for the other two strains. The simulated curves for all three strains are presents in Fig. 7.7(A) and the estimated bistable regions are all similar to those ones from bifurcation analyses, and both of them are close to the experimental results.

7.3.7 Robustness of Prediction

To evaluate the sensitivities of our predictions, we plot the estimated bistable regions for all three strains using all 200 parameter combinations shown in Fig. 7.5. As shown in Fig. 7.7(B), each point represents the bistable region generated by one parameter set with x coordinate representing the lower bound and y coordinate representing the upper bound. The lower bound of bistable region is defined as the ATc concentration corresponding to the steady state LacI concentration equals to $0.7LacI_{MAX}$ on the curve initialed from $ATc = 250ng/ml$, while the upper bound corresponds to

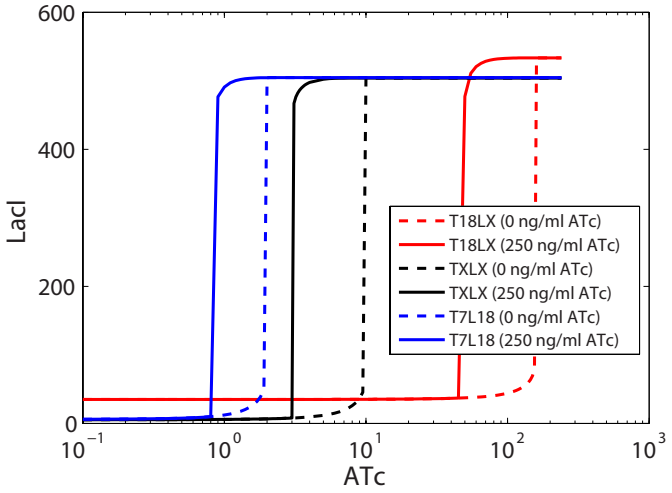

Figure 7.7: For all the three strains, simulated hysteresis curves of LacI concentration with different concentration of ATc. The dashed lines represent simulated LacI concentrations using the steady state of ATc = 0 ng/ml as initial conditions, while the solid lines represent simulated LacI concentrations using the one of ATc = 250ng/ml as initial conditions. The parameters used to calculate the bistable regions for TXLX are shown in Fig. S3, and then further transformed to parameters for T18LX and T7L18. All the simulations are calculated with ode23 solver in Matlab, and the steady state values are generated by simulating the system until $t = 10^4 s$

the one equals to $0.3LacI_{MAX}$ on the curve initialed from $ATc = 0ng/ml$. Since the parameter sets are filtered for TXLX, the numerical bistable regions (black circles) are distributed around the experimental one (black cross). For T18LX, the predicted regions (red circles) are close to the experimental one (red cross), and similarly for T7L18 (blue circles and cross). Fig. 7.7(B) shows that, with our transformation, we can predict the bistable regions for the other two strains quite well by knowing r of such strain and using the parameters estimated from those of TXLX. In addition, it can be seen that all 200 predictions are narrowly distributed, indicating robustness of our predictions.

7.3.8 Definition of Quasi-potential

For our described as 7.3, the quasi potential P at specific state variables combination (L_0, T_0) is defined as the trajectory length from (L_0, T_0) to its final stable state, as:

$$\begin{aligned} \frac{d[LacI]}{dt} &= c_{rl} + p_{e,tet}(c_{il} - c_{rl}) - \delta \cdot [LacI] \\ \frac{d[TetR]}{dt} &= c_{rt} + p_{e,lac}(c_{it} - c_{rt}) - \delta \cdot [TetR], \qquad (7.6) \\ \frac{dP}{dt} &= -\sqrt{LacI^2 + TetR^2}. \qquad (7.7) \end{aligned}$$

So P can be calculated by integrating along the evaluationary of $LacI$ and $TetR$ initialing from (L_0, T_0), as:

$$P(L_0, T_0) = \int_{t=0}^{t=\infty} [\sqrt{LacI(t)^2 + TetR(t)^2}]dt + P_S, \qquad (7.8)$$

where P_S is the quasi potential of stable static states (SSS). In system with more than one SSS, the 'depth' of each SSS, or the length of trajectory initialing very close to the unstable static state (USS) and ending up to it, is usually different. So we choose the deepest SSS as the reference point and define its quasi potential as 0, while the other 'shallow' SSS have quasi potential of C_d, which are the depth difference to the reference point.

7.4 Discussion

Combining rational engineering and natural biological regulation, here we successfully demonstrate synthetic stochastic and irreversible cell fate determination in eukaryotic cells. Bistability, stochasticity, and the resulting binary cellular decision-makings have been extensively studied [33, 34, 36, 37, 39, 41, 143, 144]. These studies illustrated both irreversible and uniform [36, 143, 144] and random yet reversible [37, 145] binary decision-makings at single cell levels. However, the case of random

and irreversible decision-making has not been studied or demonstrated. Guided by stochastic simulations on a nonlinear potential landscape, we experimentally initialized cell cultures on the separatrix of our engineered bistable system and demonstrated the first stochastic and irreversible binary cellular state determination. Independent cultures grown in the same condition but from different initial conditions showed completely different responses. This illustrates the complexity of dynamics of multistable gene networks when the effects of initial conditions and stochasticity are taken into account, more than just hysteresis.

With a more complete understanding of multistable gene networks, synthetic GRNs can serve as topological prototypes for their natural counterparts and provide novel insights not easily available through the study of natural systems. By demonstrating stochastic and irreversible cell fate determination, we have been able to shed light on the role of stochasticity in cell differentiation [146, 147]. For example, during fruit fly eye development, precursor cells in a specific area of the eye differentiate into cells with two types of photoreceptor and maintain their cell fate. It has also been reported that the differentiation of these photoreceptors are purely stochastic and independent [42]. Recent studies [43, 44] have identified a mutual inhibitory gene regulation motif as the core system driving the stochastic differentiation, but the exact mechanism of the stochastic differentiation is still unknown. The similarities between this natural system and our engineered system, both in terms of observation and underlying GRN, suggest that stochastic and irreversible cellular developments in the fruit fly eye could be due to initialization of cells near basin boundaries of a multistable network. The initialization may well be regulated by a metabolic event, like in our system, or by other mechanisms, such as epigenetics or microRNA regulation.

Finally, this work also demonstrates the power of linking synthetic GRNs to the outputs of the host cells natural GRNs. Much effort in synthetic biology has been focused on using synthetic GRNs to drive natural GRNs towards desired responses [57, 148]. Here by linking outputs of natural metabolism regulations to synthetic GRNs, we were able to realize an initial condition that is difficult to achieve through engineering alone. Such accurate initiation of the cell culture is the key to our demonstrated stochastic and random cell fate determination. This concept of harvesting natural regulatory machineries to tune synthetic GRNs will greatly increases the quality and quantity of possible perturbations that can be applied to engineered systems, hence make it possible for the engineering of future biological devices that require more sophisticated and accurate controls.

[1] DJ Watts and SH Strogatz. Collective dynamics of small-worldnetworks. *nature*, 393(6684):440–442, 1998.

[2] S Redner. How popular is your paper? an empirical study of the citation distribution. *The European Physical Journal B-Condensed Matter and Complex Systems*, 4(2):131–134, 1998.

[3] H Jeong, B Tombor, R Albert, ZN Oltvai, and A-L Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

[4] R Milo, S Shen-Orr, S Itzkovitz, N Kashtan, D Chklovskii, and U Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[5] A-L Barabási and R Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[6] AE Motter, CS Zhou, and J Kurths. Enhancing complex-network synchronization. *EPL (Europhysics Letters)*, 69(3):334, 2005.

[7] Adilson E Motter and Ying-Cheng Lai. Cascade-based attacks on complex networks. *Physical Review E*, 66(6):065102, 2002.

[8] S Boccaletti, V Latora, Y Moreno, M Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[9] XF Wang. Complex networks: topology, dynamics and synchronization. *International Journal of Bifurcation and Chaos*, 12(05):885–916, 2002.

[10] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[11] T. S. Gardner, D. di Bernardo, D. Lorenz, and J. J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301:102–105, 2003.

[12] S. Gruen, M. Diesmann, and A. Aertsen. Unitary events in multiple single neuron spiking activity. i. detection and significance. *Neu. Comp.*, 14(1):43–80, 2002.

[13] R. Gütig, A. Aertsen, and S. Rotter. Statistical significance of coincident spikes: count-based versus rate-based statistics. *Neu. Comp.*, 14(1):121–153, 2002.

[14] G. Pipa and S. Grün. Non-parametric significance estimation of joint-spike events by shuffling and resampling. *Neurocomputing*, 52:31–37, 2003.

[15] J. Bongard and H. Lipson. Automated reverse engineering of nonlinear dynamical systems. *Proc. Nat. Acad. Sci. USA*, 104:9943–9948, 2007.

[16] M. Timme. Revealing network connectivity from response dynamics. *Phys. Rev. Lett.*, 98(22):224101, 2007.

[17] D. Napoletani and T. D. Sauer. Reconstructing the topology of sparsely connected dynamical networks. *Phys. Rev. E*, 77:026103, 2008.

[18] W.-X. Wang, Q.-F. Chen, L. Huang, Y.-C. Lai, and M. Harrison. Scaling of noisy fluctuations in complex networks and applications to network prediction. *Phys. Rev. E*, 80:016116, 2009.

[19] J. Ren, W.-X. Wang, B. Li, and Y.-C. Lai. Noise bridges dynamical correlation and topology in coupled oscillator networks. *Phys. Rev. Lett.*, 104:058701, 2010.

[20] Z. Levnajić and A. Pikovsky. Network reconstruction from random phase resetting. *Phys. Rev. Lett.*, 107:034101, 2011.

[21] S. Hempel, A. Koseska, J. Kurths, and Z. Nikoloski. Inner composition alignment for inferring directed networks from short time series. *Phys. Rev. Lett.*, 107:054101, 2011.

[22] E. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Comm. Pure Appl. Math.*, 59(8):1207–1223, 2006.

[23] E. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.*, 59:1207–1223, 2005.

[24] E. Candeès and M. Wakin. An introduction to compressive sampling. *IEEE Signal Process. Mag.*, 25:21–30, 2008.

[25] D. Donoho. Compressed sensing. *IEEE Trans. Info. Theory*, 52:1289–1306, 2006.

[26] R. G. Baraniuk. Compressed sensing. *IEEE Signal Process. Mag.*, 24:118–121, 2007.

[27] W. Pan, Y. Yuan, and G.-B. Stan. Reconstruction of arbitrary biochemical reaction networks: a compressive sensing approach. In *51st Annual Conference on Decision and Control (CDC)*, pages 2334–2339. IEEE, 2012.

[28] W.-X. Wang, Y.-C. Lai, C. Grebogi, and J.-P. Ye. Network reconstruction based on evolutionary-game data via compressive sensing. *Phys. Rev. X*, 1(2):021021, 2011.

[29] W.-X. Wang, R. Yang, Y.-C. Lai, V. Kovanis, and C. Grebogi. Predicting catastrophes in nonlinear dynamical systems by compressive sensing. *Phys. Rev. Lett.*, 106(15):154101, 2011.

[30] W.-X. Wang, R. Yang, Y.-C. Lai, V. Kovanis, and M. A. F. Harrison. Time-series-based prediction of complex oscillator networks via compressive sensing. *EPL (Europhys. Lett.)*, 94(4):48006, 2011.

[31] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Info. Theory*, 52:489–509, 2006.

[32] J. E Ferrell. Self-perpetuating states in signal transduction: positive feedback, double-negative feedback and bistability. *Current opinion in cell biology*, 14(2):140–148, 2002.

[33] W. Xiong and J. E Ferrell. A positive-feedback-based bistable memory modulethat governs a cell fate decision. *Nature*, 426(6965):460–465, 2003.

[34] SR Biggar and GR Crabtree. Cell signaling can direct either binary or graded transcriptional responses. *The EMBO journal*, 20(12):3167–3176, 2001.

[35] D Nevozhay, RM Adams, E van Itallie, MR Bennett, and Balázsi G. Mapping the environmental fitness landscape of a synthetic gene circuit. *PLoS computational biology*, 8(4):e1002480, 2012.

[36] M Acar, A Becskei, and A van Oudenaarden. Enhancement of cellular memory by reducing stochastic transitions. *Nature*, 435(7039):228–232, 2005.

[37] A Becskei, B Séraphin, and L Serrano. Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion. *The EMBO journal*, 20(10):2528–2535, 2001.

[38] G Balázsi, A van Oudenaarden, and JJ Collins. Cellular decision making and biological noise: from microbes to mammals. *Cell*, 144(6):910–925, 2011.

[39] T Ellis, X Wang, and JJ Collins. Diversity-based, model-guided construction of synthetic gene networks with predicted functions. *Nature biotechnology*, 27(5):465–471, 2009.

[40] M. Louis and A. Becskei. Binary and graded responses in gene networks. *Science Signaling*, 2002(143):pe33, 2002.

[41] TS Gardner, CR Cantor, and JJ Collins. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339–342, 2000.

[42] ML Bell, JB Earl, and SG Britt. Two types of drosophila r7 photoreceptor cells are arranged randomly: A model for stochastic cell-fate determination. *Journal of Comparative Neurology*, 502(1):75–85, 2007.

[43] T Mikeladze-Dvali, MF Wernet, D Pistillo, EO Mazzoni, AA Teleman, YW Chen, S Cohen, and C Desplan. The growth regulators warts/lats and melted interact in a bistable loop to specify opposite fates in drosophila r8 photoreceptors. *Cell*, 122(5):775–787, 2005.

[44] AC Miller, H Seymour, C King, and TG Herman. Loss of seven-up from drosophila r1/r6 photoreceptors reveals a stochastic fate choice that is normally biased by notch. *Development*, 135(4):707–715, 2008.

[45] S Mukherji and A van Oudenaarden. Synthetic biology: understanding biological design from synthetic circuits. *Nature Reviews Genetics*, 10(12):859–871, 2009.

[46] MB Elowitz, AJ Levine, ED Siggia, and PS Swain. Stochastic gene expression in a single cell. *Science*, 297(5584):1183–1186, 2002.

[47] EM Ozbudak, M Thattai, I Kurtser, AD Grossman, and A van Oudenaarden. Regulation of noise in the expression of a single gene. *Nature genetics*, 31(1):69–73, 2002.

[48] A Raj, SA Rifkin, E. Andersen, and A van Oudenaarden. Variability in gene expression underlies incomplete penetrance. *Nature*, 463(7283):913–918, 2010.

[49] WJ Blake, M Kærn, CR Cantor, and JJ Collins. Noise in eukaryotic gene expression. *Nature*, 422(6932):633–637, 2003.

[50] D Nevozhay, RM Adams, KF Murphy, K Josić, and G Balázsi. Negative autoregulation linearizes the dose–response and suppresses the heterogeneity of gene expression. *Proceedings of the National Academy of Sciences*, 106(13):5123–5128, 2009.

[51] WJ Blake, G Balázsi, MA Kohanski, FJ Isaacs, KF Murphy, Y Kuang, CR Cantor, DR Walt, and JJ Collins. Phenotypic consequences of promoter-mediated transcriptional noise. *Molecular cell*, 24(6):853–865, 2006.

[52] FJ Isaacs, J Hasty, CR Cantor, and JJ Collins. Prediction and measurement of an autoregulatory genetic module. *Proceedings of the National Academy of Sciences*, 100(13):7714–7719, 2003.

[53] MB Elowitz and S Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, 2000.

[54] T Danino, O Mondragón-Palomino, L Tsimring, and J Hasty. A synchronized quorum of genetic clocks. *Nature*, 463(7279):326–330, 2010.

[55] O Mondragón-Palomino, T Danino, J Selimkhanov, L Tsimring, and J Hasty. Entrainment of a population of synthetic genetic oscillators. *Science*, 333(6047):1315–1319, 2011.

[56] J Stricker, S Cookson, MR Bennett, WH Mather, LS Tsimring, and J Hasty. A fast, robust and tunable synthetic gene oscillator. *Nature*, 456(7221):516–519, 2008.

[57] A Levskaya, OD Weiner, WA Lim, and CA Voigt. Spatiotemporal control of cell signalling using a light-switchable protein interaction. *Nature*, 461(7266):997–1001, 2009.

[58] CJ Bashor, NC Helman, S Yan, and WA Lim. Using engineered scaffold interactions to reshape map kinase pathway signaling dynamics. *Science*, 319(5869):1539–1543, 2008.

[59] L You, RS Cox, R Weiss, and FH Arnold. Programmed population control by cell–cell communication and regulated killing. *Nature*, 428(6985):868–871, 2004.

[60] FK Balagaddé, H Song, J Ozaki, CH Collins, M Barnet, FH Arnold, SR Quake, and L You. A synthetic escherichia coli predator–prey ecosystem. *Molecular systems biology*, 4(1), 2008.

[61] S Basu, Y Gerchman, CH Collins, FH Arnold, and R Weiss. A synthetic multicellular system for programmed pattern formation. *Nature*, 434(7037):1130–1134, 2005.

[62] JJ Tabor, HM Salis, ZB Simpson, AA Chevalier, A Levskaya, EM Marcotte, CA Voigt, and AD Ellington. A synthetic genetic edge detection program. *Cell*, 137(7):1272–1281, 2009.

[63] F. Takens. Detecting strange attractors in fluid turbulence. In D. Rand and L. S. Young, editors, *Dynamical Systems and Turbulence*, volume 898 of *Lecture Notes in Mathematics*, pages 366–381, Berlin, 1981. Springer-Verlag.

[64] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Phys. Rev. Lett.*, 45:712–716, 1980.

[65] H. Kantz and T. Schreiber. *Nonlinear Time Series Analysis*. Cambridge University Press, Cambridge, UK, first edition, 1997.

[66] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano. Determining lyapunov exponents from a time series. *Physica D*, 16:285–317, 1985.

[67] J. P. Eckmann, S. Oliffson Kamphorst, D. Ruelle, and S. Ciliberto. Liapunov exponents from time series. *Phys. Rev. A*, 34:4971–4979, 1986.

[68] U. Parlitz. Estimating model parameters from time series by autosynchronization. *Phys. Rev. Lett.*, 76:1232, 1996.

[69] D. Yu, M. Righero, and Lj. Kocarev. Estimating topology of networks. *Phys.l Rev. Lett.*, 97:188701, 2006.

[70] W. K.-S. Tang, M. Yu, and L. Kocarev. Identification and monitoring of biological neural network. In *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pages 2646–2649. IEEE, 2007.

[71] D. Yu and U. Parlitz. Inferring network connectivity by delayed feedback control. *PloS one*, 6:e24333, 2011.

[72] M. Timme. Revealing network connectivity from response dynamics. *Phys.l Rev. Lett.*, 98:224101, 2007.

[73] S. G. Shandilya and M. Timme. Inferring network topology from complex dynamics. *New J. Phys.*, 13:013004, 2011.

[74] Z. Levnajić and A. Pikovsky. Network reconstruction from random phase resetting. *Phys. Rev. Lett.*, 107:034101, 2011.

[75] T. Berry, F. Hamilton, N. Peixoto, and T. Sauer. Detecting connectivity changes in neuronal networks. *J. Neuros. Meth.*, 209:388–397, 2012.

[76] F. Hamilton, T. Berry, N. Peixoto, and T. Sauer. Real-time tracking of neuronal network structure using data assimilation. *Phys.l Rev. E*, 88:052715, 2013.

[77] A. Brovelli, M. Ding, A. Ledberg, Y. Chen, R. Nakamura, and S. L Bressler. Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by granger causality. *Proc. Nat. Acad. Sci. (USA)*, 101:9849–9854, 2004.

[78] D. Zhou, Y. Xiao, Y. Zhang, Z. Xu, and D. Cai. Causal and structural connectivity of pulse-coupled nonlinear networks. *Phys. Rev. Lett.*, 111:054102, 2013.

[79] O. Stetter, D. Battaglia, J. Soriano, and T. Geisel. Model-free reconstruction of excitatory neuronal connectivity from calcium imaging signals. *PLoS Comp. Bio.*, 8:e1002653, 2012.

[80] S. Hempel, A. Koseska, J. Kurths, and Z. Nikoloski. Inner composition alignment for inferring directed networks from short time series. *Phys. Rev. Lett.*, 107:054101, 2011.

[81] V. A Makarov, F. Panetsos, and O. Feo. A method for determining neural connectivity and inferring the underlying network dynamics using extracellular spike recordings. *J. Neuro. Meth.*, 144:265–279, 2005.

[82] C. Yao and E. M. Bollt. Modeling and nonlinear parameter estimation with kronecker product representation for coupled oscillators and spatiotemporal systems. *Physica D*, 227:78–99, 2007.

[83] R. Willett, R. Marcia, and J. Nichols. Compressed sensing for practical optical imaging systems: a tutorial. *Opt. Eng.*, 50:072601, 2011.

[84] A. Shabani, R. L. Kosut, M. Mohseni, H. Rabitz, M. A. Broome, M. P. Almeida, A. Fedrizzi, and A. G. White. Efficient measurement of quantum dynamics via compressive sensing. *Phys. Rev. Lett.*, 106:100401, 2011.

[85] R.-Q. Su, W.-X. Wang, and Y.-C. Lai. Detecting hidden nodes in complex networks from time series. *Phys. Rev. E*, 85(6):065201, 2012.

[86] R.-Q. Su, X. Ni, W.-X. Wang, and Y.-C. Lai. Forecasting synchronizability of complex networks from data. *Phys. Rev. E*, 85(5):056220, 2012.

[87] R.-Q. Su, Y.-C. Lai, X. Wang, and Y.-H. Do. Uncovering hidden nodes in complex networks in the presence of noise. *Sci. Rep.*, 4:3944, 2014.

[88] X. F. Wang and G. Chen. Pinning control of scale-free dynamical networks. *Physica A*, 310:521–531, 2002.

[89] X. Li, X. F. Wang, and G. Chen. Pinning a complex dynamical network to its equilibrium. *IEEE Trans. Circ. Syst. I*, 51:2074–2087, 2004.

[90] F. Sorrentino, M. di Bernardo, F. Garofalo, and G. Chen. Controllability of complex networks via pinning. *Phys. Rev. E*, 75:046103, 2007.

[91] W. Yu, G. Chen, and J. Lü. On pinning synchronization of complex dynamical networks. *Automatica*, 45:429–435, 2009.

[92] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physio.*, 17:500–544, 1952.

[93] R. FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophys. J.*, 1:445–466, 1961.

[94] J. Nagumo, S. Arimoto, and S. Yoshizawa. An active pulse transmission line simulating nerve axon. *Proc. IRE*, 50:2061–2070, 1962.

[95] E. Ott. *Chaos in Dynamical Systems*. Cambridge University Press, Cambridge, UK, second edition, 2002.

[96] P. W. Anderson. More is different: broken symmetry and the nature of the hierarchical structure of science. *Science*, 177:393–96, 1972.

[97] S. Strogatz. *Sync: The emerging science of spontaneous order*. Hyperion, 2003.

[98] A. Pikovsky, M. Rosenblum, J. Kurths, and R. C Hilborn. *Synchronization: A universal concept in nonlinear sciences*, volume 2. Cambridge University Press Cambridge, 2002.

[99] L. M Pecora and T. L Carroll. Synchronization in chaotic systems. *Physical review letters*, 64(8):821, 1990.

[100] L. M Pecora and T. L Carroll. Master stability functions for synchronized coupled systems. *Physical Review Letters*, 80(10):2109, 1998.

[101] L. Huang, Q. Chen, Y.-C. Lai, and L. M Pecora. Generic behavior of master-stability functions in coupled nonlinear dynamical systems. *Physical Review E*, 80(3):036204, 2009.

[102] MK S. Yeung, J. Tegnér, and J. J Collins. Reverse engineering gene networks using singular value decomposition and robust regression. *Proceedings of the National Academy of Sciences*, 99(9):6163–6168, 2002.

[103] D. di Bernardo, M. J. Thompson, T. S. Gardner, S. E. Chobot, E. L. Eastwood, A. P. Wojtovich, S. J. Elliott, S. E. Schaus, and J. J. Collins. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature biotechnology*, 23(3):377–383, 2005.

[104] S. G. Shandilya and M. Timme. Inferring network topology from complex dynamics. *New J. Phys.*, 13:013004, 2011.

[105] P. Erdös and A. Rényi. On random graphs i. *Publ. Math. Debrecen*, 6:290–291, 1959.

[106] E. N Lorenz. Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, 20(2):130–141, 1963.

[107] M. Hénon. A two-dimensional mapping with a strange attractor. *Communications in Mathematical Physics*, 50(1):69–77, 1976.

[108] JD Farmer and JJ Sidorowich. Predicting chaotic time series. *Physical review letters*, 59(8):845, 1987.

[109] TD Sauer. Reconstruction of shared nonlinear dynamics in a network. *Physical review letters*, 93(19):198701, 2004.

[110] MA Nowak. *Evolutionary dynamics*. Harvard University Press, 2006.

[111] G Szabó and G Fath. Evolutionary games on graphs. *Physics Reports*, 446(4):97–216, 2007.

[112] E. Sontag. Network reconstruction based on steady-state data. *Essays Biochem.*, 45:161–176, 2008.

[113] Y. Yuan, G.-B. Stan, S. Warnick, and J. Goncalves. Robust dynamical network reconstruction. In *49th IEEE Conference on Decision and Control (CDC)*, pages 810–815. IEEE, 2010.

[114] E. Candeès. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1433–1452. Madrid, Spain, 2006.

[115] OE Rössler. An equation for continuous chaos. *Physics Letters A*, 57(5):397–398, 1976.

[116] EJ Candes, MB Wakin, and SP Boyd. Enhancing sparsity by reweighted l-1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.

[117] A. Brovelli, M. Ding, A. Ledberg, Y. Chen, R. Nakamura, and S. L Bressler. Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by granger causality. *Proc. Nat. Acad. Sci. USA*, 101(26):9849–9854, 2004.

[118] W. K.-S. Tang, M. Yu, and L. Kocarev. Identification and monitoring of biological neural network. In *International Symposium on Circuits and Systems*, pages 2646–2649. IEEE, 2007.

[119] D. Yu and U. Parlitz. Inferring network connectivity by delayed feedback control. *PloS ONE*, 6:e24333, 2011.

[120] W.-X. Wang, J. Ren, Y.-C. Lai, and B. Li. Reverse engineering of complex dynamical networks in the presence of time-delayed interactions based on noisy time series. *Chaos*, 22:033131, 2012.

[121] T. Berry, F. Hamilton, N. Peixoto, and T. Sauer. Detecting connectivity changes in neuronal networks. *J. Neurosci. Meth.*, 209(2):388–397, 2012.

[122] O. Stetter, D. Battaglia, J. Soriano, and T. Geisel. Model-free reconstruction of excitatory neuronal connectivity from calcium imaging signals. *PLoS Comp. Biol.*, 8(8):e1002653, 2012.

[123] F. Hamilton, T. Berry, N. Peixoto, and T. Sauer. Real-time tracking of neuronal network structure using data assimilation. *Phys. Rev. E*, 88(5):052715, 2013.

[124] D. Zhou, Y. Xiao, Y. Zhang, Z. Xu, and D. Cai. Causal and structural connectivity of pulse-coupled nonlinear networks. *Phys. Rev. Lett.*, 111(5):054102, 2013.

[125] M. Timme and J. Casadiego. Revealing networks from dynamics: an introduction. *J. Phys. A. Math. Theo.*, 47:343001, 2014.

[126] V. A Makarov, F. Panetsos, and O. Feo. A method for determining neural connectivity and inferring the underlying network dynamics using extracellular spike recordings. *J. Neurosci. Meth.*, 144(2):265–279, 2005.

[127] Z.-S. Shen, W.-X. Wang, Y. Fan, Z.-R. Di, and Y.-C. Lai. Reconstructing propagation networks with natural diversity and identifying hidden source. *Nat. Comm.*, 5:4323, 2014.

[128] Xiaoqun Wu. Synchronization-based topology identification of weighted general complex dynamical networks with time-varying coupling delay. *Physica A*, 387(4):997–1008, 2008.

[129] D. Husmeier. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic bayesian networks. *Bioinformatics*, 19(17):2271–2282, 2003.

[130] A.H. Sayed, A. Tarighat, and N. Khajehnouri. Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Process. Mag.*, 22(4):24–40, 2005.

[131] I.-C. Moon and K.M. Carley. Modeling and simulating terrorist networks in social and geospatial dimensions. *IEEE Intell. Syst.*, 22(5):40–49, 2007.

[132] C. Li and G. Chen. Synchronization in general complex dynamical networks with coupling delays. *Physica A*, 343:263–278, 2004.

[133] M. Dhamala, V.K. Jirsa, and M. Ding. Enhancement of neural synchrony by time delay. *Physical review letters*, 92(7):074104, 2004.

[134] A. Bellen and M. Zennaro. *Numerical methods for delay differential equations*. Oxford University Press, 2013.

[135] Y. Shang and W. Ruml. Improved mds-based localization. In *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2640–2651. IEEE, 2004.

[136] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz. Localization from mere connectivity. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 201–212. ACM, 2003.

[137] Sudin Bhattacharya, Qiang Zhang, and Melvin E Andersen. A deterministic map of waddington's epigenetic landscape for cell fate specification. *BMC Syst. Biol.*, 5(1):85, 2011.

[138] KY Kim and J Wang. Potential energy landscape and robustness of a gene regulatory network: toggle switch. *PLoS computational biology*, 3(3):e60, 2007.

[139] SH Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Westview press, 2014.

[140] DT Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of computational physics*, 22(4):403–434, 1976.

[141] M Johnston. A model fungal gene regulatory mechanism: the gal genes of saccharomyces cerevisiae. *Microbiological reviews*, 51(4):458, 1987.

[142] RS Sikorski and P Hieter. A system of shuttle vectors and yeast host strains designed for efficient manipulation of dna in saccharomyces cerevisiae. *Genetics*, 122(1):19–27, 1989.

[143] G. Yao, TJ Lee, S Mori, JR Nevins, and L You. A bistable rb–e2f switch underlies the restriction point. *Nature cell biology*, 10(4):476–482, 2008.

[144] CM Ajo-Franklin, DA Drubin, JA Eskin, EPS Gee, D Landgraf, I Phillips, and PA Silver. Rational design of memory in eukaryotic cells. *Genes & development*, 21(18):2271–2276, 2007.

[145] M Kærn, TC Elston, WJ Blake, and JJ Collins. Stochasticity in gene expression: from theories to phenotypes. *Nature Reviews Genetics*, 6(6):451–464, 2005.

[146] JM Pedraza and A van Oudenaarden. Noise propagation in gene networks. *Science*, 307(5717):1965–1969, 2005.

[147] A Raj and A van Oudenaarden. Nature, nurture, or chance: stochastic gene expression and its consequences. *Cell*, 135(2):216–226, 2008.

[148] A Prindle, P Samayoa, I Razinkov, T Danino, LS Tsimring, and J Hasty. A sensing array of radically coupled genetic/biopixels/'. *Nature*, 481(7379):39–44, 2012.

APPENDIX A

ACHIEVEMENTS DURING PHD STUDIES

Following are the relevant works in this dissertation.

1. R.-Q. Su, X. Ni, W.-X. Wang, and Y.-C. Lai. Forecasting synchronizability of complex networks from data. *Phys. Rev. E*, 85(5):056220, 2012.

2. R.-Q. Su, W.-X. Wang, and Y.-C. Lai. Detecting hidden nodes in complex networks from time series. *Phys. Rev. E*, 85(6):065201, 2012.

3. M. Wu, R.-Q. Su, X.-H. Li, T. Ellis, Y.-C. Lai, and X. Wang. Engineering of regulated stochastic cell fate determination. *Proceedings of the National Academy of Sciences*, 110(26):10610–10615, 2013.

4. R.-Q. Su, Y.-C. Lai, X. Wang, and Y.-H. Do. Uncovering hidden nodes in complex networks in the presence of noise. *Sci. Rep.*, 4:3944, 2014.

5. R.-Q. Su, Y.-C. Lai, and Xiao Wang. Identifying chaotic FitzHughNagumo neurons using compressive sensing. *Entropy*, 16(7):3889–3902, 2014.

6. R.-Q. Su, W.-X. Wang, X. Wang, and Y.-C. Lai. Data based reconstruction of complex geospatial networks, nodal positioning, and detection of hidden node. *submitted*.

Other work that has not been included in this dissertation are listed below.

7. L.-Z. Wang, R.-Q. Su, Z.-G. Huang, X. Wang, W.-X. Wang, C Grebogi and Y.-C. Lai. Control and controllability of nonlinear dynamical networks: a geometrical approach. *arXiv.*, 1509.07038, 2015.

Ri-Qi Su was born in 1986 in Guangxi, China. He received a B. S. in Physics from the University of Science and Technology of China(USTC) in 2008 and a M. S. in Physics from the University of Science and Technology of China in 2010. While in USTC, he worked with Prof. Bing-Hong Wang on nonlinear dynamics and complex networks. He began graduate study at the Arizona State University in the spring of 2011 on constructing and controlling nonlinear systems with his advisor Prof. Ying-Cheng Lai and Prof. Xiao Wang.