Network Inference Using Information-theoretical Approaches, Statistical Hypothesis Tests, and The Amplitude-Phase Modulated Surrogate Data Method

Hüseyin YILDIRIM

A thesis submitted for the degree of

Doctor of Philosophy

School of Mathematics, Statistics and Actuarial Science
University of Essex

Date of submission for examination: April 2024

Network Inference using information-theoretical approaches, statistical hypothesis tests, and the amplitude-phase modulated surrogate data method

Abstract

The last few decades have witnessed the emergence of network science aimed at modelling natural phenomena across a diverse spectrum of disciplines, including biology, social sciences, and neuroscience, among others. One of the notable advances in this field is the concept of mutual information rate (MIR), which quantifies the information flow per unit of time between different components (nodes) within a network. The MIR framework holds the potential to reveal connectivity patterns in complex networks using time-series data. A critical challenge in using MIR is the establishment of appropriate thresholds for successful network inference. We propose a new method to infer connectivity in networks using MIR, statistical tests and amplitude-phase modulated surrogate data (APMSD). The method uses MIR and statistical hypothesis tests to infer network connectivity, introducing a new method to generate surrogate data, which removes the correlation of amplitude and synchronisation of the phases in the recorded signals, by randomising their instantaneous amplitudes and phases. The APMSD method compares MIRs between the pairs of nodes of the data from the coupled or stochastic models with those of the APMSD generated from the data randomising instantaneous amplitudes and/or phases. We discuss the mathematical aspects of the APMSD method and present numerical results for Gaussian-distributed correlated data, networks of coupled maps and continuous deterministic systems, the stochastic Kuramoto system, and for dynamics on heterogeneous networks. The importance of our method stems from the analytic signal concept, introduced by Gabor in 1946 and the Hilbert transform, as it provides us with the quantification of the contribution of amplitude correlation (linear or nonlinear) and phase synchronisation in the connectivity among nodes within a network. Our method shows great potential for recovering the network structure in coupled deterministic and stochastic systems and in heterogeneous networks with weighted connectivity.

To my beloved parents,

Süreyya & Bayram

Declaration

The work in this thesis is based on research carried out with my supervisor Dr. Chris G. Antonopoulos in the School of Mathematics, Statistics and Actuarial Science, University of Essex, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification, and it is all my own work, unless referenced, in the text.

Hüseyin Yıldırım
2024

Acknowledgements

First and foremost, I would like to express my heartfelt appreciation to my dedicated supervisor, Dr. Chris Antonopoulos, for his invaluable guidance, support, and encouragement throughout my research journey. His expertise, patience, and unwavering commitment played a pivotal role in shaping this thesis. I am also thankful to Dr. Yassir Rabhi, Chair of the Supervisory Panel Board, for his insightful feedback.

I am deeply grateful to all those involved with the School of Mathematics, Statistics, and Actuarial Science at the University of Essex. Special thanks go to Julie Pierson and Sanja Marjanovic for their diligent efforts in ensuring smooth operations. I feel privileged to have had the opportunity to engage, collaborate, and learn from the esteemed academic staff, particularly Dr. Osama Mahmoud, Dr. Yanchun Bao, Professor Chris Saker, Dr. Dan Brawn, Dr. Xu Chen, and Dr. Murat Akman. In addition, I am grateful for the unexpected encounters with the exceptional research community, including Dr. Zulkarnain, Anthony Miller, Nora Alqwaizany, Alex Partner, Sarah Alsaed, and Anusa Suwanwong.

My heartfelt thanks go to all friends who made my stay in Colchester more enjoyable, including Salih, Mustafa, Nazanin, Cemre, Annika, Weiyong, and Cathal. I also extend my gratitude to friends who are far away but always give me their support, including Seçkin, Sercan, and Elif Burcu.

I would like to express my profound gratitude to my parents and brothers, Ali Arda and Hilmi, for their unending love, encouragement, and understanding throughout this journey. Their unwavering support has been my cornerstone of strength and motivation.

I extend my sincere appreciation to the Republic of Türkiye Ministry of National Education for providing financial support, which allowed me to pursue my academic endeavours and complete this thesis.

Last but not least, I would like to express my heartfelt gratitude to Sevde Büşra Bayrak for her invaluable partnership, friendship, steadfast support, and willingness to engage in discussions about the topics addressed within the thesis.

Publications

- Part of the work in chapter 7 has been published in: Hüseyin Yıldırım, and Chris G. Antonopoulos. Network inference using mutual information rate, statistical tests and amplitude-phase modulated surrogate data. *Chaos, Solitons & Fractals*, 188: 115554, 2024.
- All Matlab code used in this work is available on GitHub: `https://github.com/hy20577/PhD_thesis_all_codes`

Contents

List of Figures

List of Tables

Acronyms

AB4 4^{th}-order Adams-Bashford.

APMSD amplitude-phase modulated surrogate data.

AS almost synchronisation.

BA Barabási-Albert.

CDT correlation decay time.

CS complete synchronisation.

EEG electroencephalogram.

ER Erdös-Renyi.

FDR false discovery rate.

FPR false positive rate.

FT Fourier transform.

GS generalised synchronisation.

HR Hindmarsh-Rose.

HT Hilbert transform.

ILS intermittent lag synchronisation.

IPS imperfect phase synchronisation.

LS lag synchronisation.

MI mutual information.

MIR mutual information rate.

MLE maximum Lyapunov exponent.

ODE ordinary differential equation.

PC Pearson correlation.

PS phase synchronisation.

RK23 2^{nd}-3^{rd} order Runge-Kutta.

RK4 4^{th}-order Runge-Kutta.

RK45 4^{th}-5^{th} order Runge-Kutta.

RL ring lattice.

RSD random surrogate data.

SDE stochastic differential equation.

SF scale-free.

SW small-world.

TPR true positive rate.

TWSD twin surrogate data.

WS Watts-Strogatz.

Chapter 1

Introduction

Complex network theory focuses on the study of relationships among units in a system, which can be revealed by studying their temporal evolution. These systems often exhibit highly diverse dynamical behaviours, such as sensitivity to initial conditions and chaotic behaviour [63]. In a network of interconnected units, nodes represent the units of the system, and links the interactions among them. With the increase in computing power, the last few decades have witnessed the rise of complex network science, aimed at modelling natural phenomena [16].

Network inference has been extensively studied in various fields [1, 16] including, ecology [58], biology [32,48], engineering [47], finance [26,48], neuroscience [50,54], social sciences [29], epidemiology [40], etc., using correlation methods [6], information-theoretical approaches [2, 3, 12, 26], phase measures [50], regression-based methods [58], probabilistic methods [54], and dynamical models [48] to name a few. In their research, the authors in [58] explored trophic and nontrophic relationships between species, investigating their impact on population size. Gene regulatory networks can indicate the interaction between genes, and understanding human biology has potential to improve personalised treatment [33]. The authors in [47] presented recent findings in the study of complex power grid networks, which are crucial to society, as they enable the efficient distribution of the electricity infrastructure. Examining the interaction between a country's currency and the stock market, the authors in [26] presented valuable insights for portfolio management. Studying the functional network of the brain through intracranial electroencephalogram (EEG) data, the work in [50] contributed new information on the interaction between dorsal and ventral visual streams, strongly linked to human cognition. Introducing a unified Bayesian inference framework, [29] proposed a method to investigate animal social networks from observational data. Comparing the Dynamics Bayesian Network, Lasso Regression, and Pearson correlation coefficient methods on species' presence-absence data, the authors in [58] concluded that this type of data may not always provide sufficient information to reveal interactions. Consequently, they emphasised the need for cautious interpretation of inferred networks. The authors in [40] simulate various scenarios related to the spread of COVID-19 using network theory.

Exploring interactions between system units across disciplines often involves the use of networks, a well-studied domain that employs various mathematical approaches [44]. The authors in [6] introduced an approach that combines correlation with prior knowledge to select an appropriate cutoff point in network inference, demonstrating its implementation in untargeted metabolomics and transcriptomics data. Studies in [3, 12, 60] delved into the properties of MIR, emphasising its efficacy in capturing nonlinear relations in time-series data compared to Pearson correlation. Despite a slightly higher computational cost, information-theoretical methods, as discussed in [33], outperform the Pearson correlation in also capturing nonlinear interdependencies. In brain network analysis, [50] uses phase locking values as a symmetric measure to construct undirected networks and the directed transfer function to investigate causal interactions between brain regions from intracranial EEG data. Although Bayesian network analysis focusses on directed acyclic graphs, overlooking self-loop relations, it remains crucial in some research areas [33]. To address this limitation, the Dynamic Bayesian Network approach considers the probability distribution of data. In [48], the authors improved the DYNOTEARS algorithm using dynamic Bayesian networks on time series data, demonstrating its application on synthetic and real data from finance and molecular biology. Despite significant progress in the field of network inference, there are still many open questions.

One of the notable advances in the field of network inference is MIR, which quantifies the flow of information per unit of time among different nodes in a network [3, 12]. As was shown, MIR can successfully infer the structure in various types of networks and dynamics [12] and real datasets [26]. However, a major challenge in using MIR is the definition of an appropriate threshold for successful connectivity inference. The authors in [62], implemented mutual information-based tools to investigate differences in brain connectivity in post-stroke patients with different levels of depression. They considered that 10–30 % pairs that have maximum mutual information are connected, leading to various network topologies. The authors in [26], introduced a way to address the problem of defining a proper threshold (thresholding problem), where they proposed the use of an additional directed link to help them infer the interdependencies between stock indices and financial markets of countries around the world. However, this involves the use of additional data from chaotic logistic maps to compare with the recorded data, which brings about the problem of choosing a system or why choosing a system and not using the dataset itself to define a threshold. In light of this, the author in [2] proposes another approach to tackle the thresholding problem: the use of statistical tests based on surrogate data. This approach considers the source of connectivity as a linear or phase relation among pairs of nodes by comparing the MIR values of the original data with the MIR values of random- or twin-surrogate data, which destroy all linear or phase relations in the signals. However, this does not take into account the different contributions of amplitude correlation and phase synchronisation in the signals to network connectivity. In addition, it does not provide information on which surrogate method to use. To overcome these limitations, in this thesis, we introduce a new method to generate surrogate data, called the APMSD method, which removes correlation and phase synchronisation by randomising amplitudes and instantaneous phases in the recorded signals individually. The method is based on the computation of MIR and statistical hypotheses tests to infer network connectivity [2] using APMSD. The proposed method compares MIR between pairs of signals from the network with the MIR values of pairs of APMSD coming from the signals. In particular, the APMSD method is based on [13, 14], which provides detailed information on the extraction of phase measures from signals, on [15, 16] which discusses how to use the Gabor transform to convert signals into their analytic signal using the Hilbert transform. This enables one to compute the instantaneous amplitudes and phases of a signal. Past studies have revealed that signal dependencies are influenced by the relationships between amplitudes and phases [2, 16]. Building upon this, our method randomises both the amplitudes and the phases using two percentages, denoted by pc_1 and pc_2 in the paper, respectively. We show that for all systems and networks studied, the method was able to find at least one pair of randomisation parameters that leads to perfect network inference. The novelty of our method is that it can quantify the contribution of amplitude correlation and phase synchronisation in network connectivity.

This thesis is organised as follows:

Chapter 2 introduces the preliminary concepts of network theory, forming the basis for the subsequent chapters. We begin with a historical problem in graph theory: the Königsberg bridge, followed by a brief summary of key structural properties of undirected graphs, including node degree distribution, diameter, and clustering coefficient. Finally, we generate different types of networks, such as ring lattice, Erdös-Renyi (ER) random networks, Barabási-Albert (BA) small-world networks, and Watts-Strogatz (WS) scale-free networks, to discuss their structural properties and use them in network inference in Chapter 5.

Chapter 3 is dedicated to the theoretical framework of nonlinear dynamical systems and chaos theory. In this context, we discuss the numerical integration methods for ordinary differential equations (ODEs), stability of the deterministic systems, and the concept of Lyapunov exponents. Additionally, we introduce the discrete time coupled deterministic systems such as logistic and circle maps, continuous time deterministic systems such as Hindmarsh-Rose (HR), Lorenz systems and Kuramoto phase oscillators and a stochastic system, Kuramoto system with Wiener process. We thoroughly analyse the dynamics of these systems, laying the groundwork for their subsequent application in network inference.

In Chapter 4, we discuss the concept of synchronisation in chaotic systems. Even though it may seem controversial, as chaotic systems are sensitive to initial conditions and synchrony implies harmony between units, it does occur under some coupling configuration. In this context, we discuss the master-slave and bidirectionally coupled systems. To analyse the phase synchronisation of chaotic trajectories, we decompose the signal using the concept of analytic signal introduced by Gabor [23] using Hilbert transform (HT). Furthermore, we present two examples for complete synchronisation (CS), phase synchronisation (PS) and imperfect phase synchronisation (IPS) in the context of a network of 5 nodes. In this context, to quantify the amplitude correlation and phase synchronisation between nodes in a network, we use three measures: (1) Pearson correlation (PC), (2) Kuramoto phase order [37], and (3) pairwise phase order or pairwise phase synchronisation [27].

Chapter 5 is dedicated to discussing the information theoretical approaches in the context of network inference. Firstly, we compare mutual information with Pearson correlation in a quick example to encourage the reader. In this context, we show that mutual information captures both nonlinear and linear relations, whereas PC captures only linear relations. We continued discussing the theory of the information-theoretical approaches with important theorems and their proofs. We also introduced double normalised MIR, time lag versions of PC, MIR and double normalised MIR. Next, we present the results of the methods that implement the various dynamics, including Gaussian-distributed data, coupled logistic and circle maps, HR and the Lorenz systems. Furthermore, to investigate the effect of synchronisation, we implement MIR on two data sets providing CS, PS and IPS discussed in Chapter 4.

To address the thresholding problem, the author in [2] suggests hypothesis testing using MIR and surrogate data including random surrogate data (RSD) and twin surrogate data (TWSD). In Chapter 6, we discuss the theoretical framework of the method employing it in various dynamics and networks, including Gaussian distributed correlated data, deterministic discrete and continuous systems, and stochastic dynamical systems. The results show that using RSD and TWSD does not allow to infer the initial network structure successfully in many cases.

Therefore, to improve the performance of statistical tests method, we propose the hypothesis testing using amplitude-phase modulated surrogate data. Chapter 7 is dedicated to discussing the theory of theAPMSD method and its applications on various dynamics and network topologies, including the systems discussed in Chapter 6 for comparison reasons. The APMSD method shows great potential in recovering the network structure in coupled deterministic and stochastic systems and in heterogeneous networks with weighted connectivity.

Finally, we conclude our study by establishing connections among the results discussed in the thesis and by stating possible extensions of this study in Chapter 8.

Chapter 2

Brief Introduction to Network Theory

2.1 From Graph Theory to Networks

The history of graph theory in mathematical problem solving dates back to 1739 [4]. Swiss-born mathematician Leonhard Euler had a significant influence on various mathematical concepts. Although Euler is known for his groundbreaking equation $e^{i\pi} = -1$, which is a special case of the operation of two irrational and one complex numbers equating to an integer, his work also played a role in the birth of graph theory.

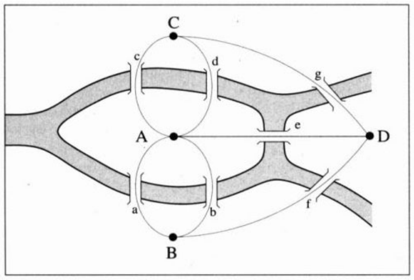

Figure 2.1: The graphical representation of the seven bridges of Könisberg in St. Petersburg, where uppercase letters denote vertices and lowercase letters connecting lines, referred to as edges, what is called a graph in mathematics.

During his time in St. Petersburg, Euler encountered a famous question among the locals: Could one visit all the islands using each bridge exactly once? This problem involved four islands connected by seven bridges (see Fig. 2.1). Many intellectuals grappled with this challenge for days during that era. Euler reformulated the question as a graphical problem. He assigned uppercase letters A, B, C and D to the islands (referred to as vertices or nodes) and lowercase letters a, b, c, d, e, f and g to the bridges (referred to as edges or links). In 1736, he proposed a solution, proving that there was no way to traverse all islands while crossing each bridge only once.

Euler's solution was both simple and elegant: he counted the number of bridges connected to each island. For example, there were three bridges connecting islands B (or C) to others. This meant that the degrees of nodes A, B, C, and D were 5, 3, 3, and 3, respectively. Although it might have been possible for only two nodes with odd degrees (representing the start and end points) to visit every island by crossing each bridge once, the question at hand involved four nodes with odd degrees. Therefore, there is no way to visit all the islands by crossing the bridges once. This brings us to the mathematical definition of a graph, given below [18]

Definition 2.1.1 (Graph)**.** *A graph G is an ordered triple ($V(G)$, $E(G)$, ψ_G) consisting of a non-empty set $V(G)$ of vertices, a set $E(G)$, disjoint from $V(G)$, of edges, and an incidence function ψ_G that associates with each edge of G an unordered pair of (not necessarily distinct) vertices of G. If e is an edge and u and v are vertices such that $\psi_G = uv$, then e is said to join u and v; the vertices u and v are called the ends of e.*

We can define the components of the graph, given in the definition, for the problem of Könisberg Bridge by,

$$\begin{aligned}
G &= (V(G), E(G), \psi_G), \quad \text{where} \\
V(G) &= \{A, B, C, D\}, \\
E(G) &= \{a, b, c, d, e, f, g\}, \\
\psi_G(a) &= AB, \psi_G(b) = AB, \psi_G(c) = AC, \psi_G(d) = AC, \\
\psi_G(e) &= AD, \psi_G(f) = BD, \psi_G(g) = CD.
\end{aligned}$$

The authors in [52] discussed the distinction between the terms “Graph” and “Network” that they are commonly used interchangeably in the scientific literature. However, the difference lies in that the graph refers to the schematic visualisation to solve a mathematical problem, whereas the network refers to graphical representation of real-life problem such as transportation maps, social interactions, protein-protein interactions, etc. Despite their close relationship, we shall henceforth refer to them as “network” and their components, vertices and edges, as “nodes” and “links”, respectively.

Networks have been extensively used in various fields [1,16], including ecology [58], biology [32, 48], finance [26, 48], neuroscience [50, 54], engineering [47], epidemiology [40]. In their research, the authors in [58] explored trophic and nontrophic relationships among species, investigating their impact on population size. Gene regulatory networks can indicate the interaction between genes, and understanding human biology has potential to improve personalised treatment [33]. The network of stock markets in different countries might provide a useful tool for portfolio managers. Studying the functional network of the brain through intracranial EEG data, [50] contributed new information on the interaction between dorsal and ventral visual streams, strongly linked to human cognition. The authors in [47] presented recent findings in the study of complex power grid networks, which are crucial to society, as they enable the efficient distribution of the electricity infrastructure. The authors in [40] simulate various scenarios related to the spread of COVID-19 using network theory. In conclusion, the study of networks spans a wide range of disciplines, highlighting their importance and versatility.

There are various ways to represent networks; they can be directed such as the websites traffic in World Wide Web (www) or undirected such as friendship on Facebook. In the network of www, one visits a website from another, which constitutes a directed relation. However, when A is a friend of B, it implies that B is also a friend of A, indicating an undirected relationship. Networks can also feature multiple links between any pair of nodes, as seen in the Königsberg Bridge problem, or they can contain self-links, representing instances such as self-citations of an author in a citation network or a protein capable of interacting with itself. However, in this research, we focus on undirected networks with no multiple links and no self-loops, referred to as simple networks. Therefore, the ensuing discussions will be confined to this type of network only.

This chapter is dedicated to presenting the structural attributes of networks, including density, node degree, graph Laplacian, diameter, and clustering coefficient. Additionally, it aims to introduce the most well-known network types, such as ring lattices, random graphs, small-world networks, and scale-free networks.

2.2 Structure of Networks

In this section, we will discuss the structural properties of networks using concepts from graph theory, such as the adjacency matrix, the weighted adjacency matrix, the density, degree of node, the Laplacian graph, the path length, diameter and clustering coefficient using the networks given in Fig. 2.2.

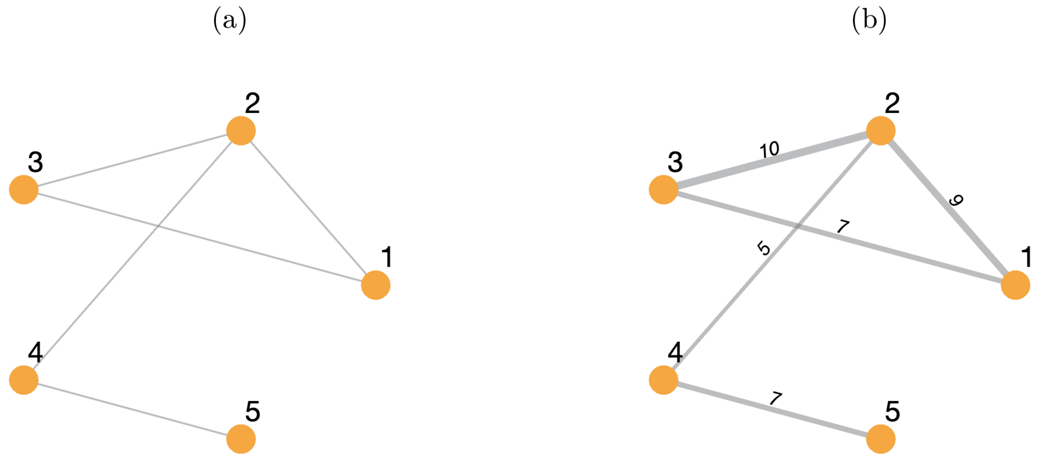

Figure 2.2: **Examples of a simple and a weighted network.** (a) A simple network of 5 nodes and 5 links, (b) a weighted network of 5 nodes and 5 links, where their widths denote the weights of connectivity. Note the numbers on the links denote the weights of connectivity, which is proportional to the widths of the links.

2.2.1 The Adjacency Matrix and Density of a Network

For a simple network of N nodes, the adjacency matrix is a $N \times N$ matrix with 0s and 1s as entries, for unconnected and connected pairs of nodes, respectively, and is given by

$$\mathbf{A}_{N\times N} = \begin{cases} \mathbf{A}_{ij} = 1, & \text{if nodes } i \text{ and } j \text{ are connected,} \\ \mathbf{A}_{ij} = 0, & \text{if nodes } i \text{ and } j \text{ are unconnected.} \end{cases}$$

For the given simple network in Fig. 2.2(a), we can construct its adjacency matrix,

$$\mathbf{A}_{5\times 5} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

The *density* of a network can be defined as the ratio of the number of existing links to all possible links in a fully-connected network, where all nodes are connected to each other. For example, the density of the network in Fig. 2.2(a) is 0.5 as the number of links is 5 and the maximum number of links in a fully-connected network of 5 nodes is 10.

The weighted adjacency matrix of a weighted network is given by,

$$\mathbf{W}_{N\times N} = \begin{cases} \mathbf{W}_{ij} = w_{ij}, & \text{if nodes } i \text{ and } j \text{ are connected} \\ \mathbf{W}_{ij} = 0, & \text{if nodes } i \text{ and } j \text{ are unconnected,} \end{cases}$$

where w_{ij} is the weight of the link between nodes i and j. For the network in Fig. 2.2(b), its weighted adjacency matrix is

$$\mathbf{W}_{5\times 5} = \begin{bmatrix} 0 & 9 & 7 & 0 & 0 \\ 9 & 0 & 10 & 5 & 0 \\ 7 & 10 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 7 \\ 0 & 0 & 0 & 7 & 0 \end{bmatrix}.$$

2.2.2 Node Degree and Degree Distributions

The node degree of a node v_i, is defined as the number of its neighbours and is denoted by $deg(v_i)$. It can be computed from its adjacency matrix, $\mathbf{A}$, by

$$deg(v_i) = \sum_{j=1}^{N} \mathbf{A}_{ij}.$$

For example, we can compute the node degrees of the network in Fig. 2.2(a) as follows

$$\begin{aligned} deg(v_1) = deg(v_3) = deg(v_4) &= 2, \\ deg(v_2) &= 3, \\ deg(v_5) &= 1. \end{aligned}$$

The node degree matrix, $\mathbf{D}$, is diagonal, its diagonal elements are the node degrees, and is given by

$$\mathbf{D} = \begin{bmatrix} k_1 & 0 & \dots & 0 \\ 0 & k_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & k_N \end{bmatrix}, \tag{2.1}$$

where $k_i = deg(v_i)$. For example, the node-degree matrix of the network in Fig. 2.2(a) is

$$\mathbf{D}_{5\times5} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.2}$$

Moreover, the probability distribution of the node degrees of a network can be computed by the probabilities,

$$p(k) = \frac{N_k}{N},$$

where, $p(k)$ is the probability of nodes with degree k to exist. In particular, N_k is the number of nodes with degree k and N is the total number of nodes in the network (or the so-called network size). For example, the probability distribution of degrees of the network in Fig. 2.2(a) are $p(1) = \frac{1}{5}$, $p(2) = \frac{3}{5}$, $p(3) = \frac{1}{5}$ and $p(4) = 0$. As expected, their sum is equal to 1. To characterise the large-size networks that we discuss later in Sec. 2.3, the probability distribution of degrees will be crucial.

2.2.3 The graph Laplacian

In statistical mechanics, particle diffusion can be thought of as the movement of gas particles from high- to low-pressure areas. In network theory, diffusion refers to the amount of a quantity that moves along a path between nodes i and j. Consequently, the diffusion rate is the change of quantity, ψ_i in a small amount of time and is given by

$$\frac{d\psi_i}{dt} = C \sum_j \mathbf{A}_{ij}(\psi_j - \psi_i). \tag{2.3}$$

In Eq. (2.3), for the constant C (is also called diffusion constant), we can split it two terms:

$$\begin{aligned}\frac{d\psi_i}{dt} &= C\sum_j \mathbf{A}_{ij}\psi_j - C\psi_i\sum_j \mathbf{A}_{ij} = C\sum_j \mathbf{A}_{ij}\psi_j - C\psi_i k_i \\ &= C\sum_j (\mathbf{A}_{ij} - \delta_{ij}k_i)\psi_j, \end{aligned} \tag{2.4}$$

where $k_i = \sum_j \mathbf{A}_{ij}$ is the degree of node i, δ_{ij} the Kronecker delta, which is 1 for $i = j$ and 0 otherwise. We can express Eq. (2.4) in matrix form by

$$\frac{d\psi}{dt} = C(\mathbf{A} - \mathbf{D})\psi, \tag{2.5}$$

where ψ is the vector of quantities ψ_i and $\mathbf{D}$ the degree diagonal matrix in Eq. (2.1). Looking at Eq. (2.5), we can define the graph Laplacian, $\mathbf{L}$ of the adjacency matrix, $\mathbf{A}$ of the network by

$$\mathbf{L} = \mathbf{D} - \mathbf{A}. \tag{2.6}$$

Then, the differential equation in Eq. (2.5) becomes

$$\frac{d\psi}{dt} + C\mathbf{L}\psi = 0.$$

In particular, computing $\mathbf{L}$ from Eq. (2.6), one arrives at

$$\mathbf{L}_{ij} = \begin{cases} k_{ij}, & \text{if } i = j, \\ -1, & \text{if } i \neq j \text{ and nodes } i \text{ and } j \text{ are connected,} \\ 0, & \text{otherwise.} \end{cases} \tag{2.7}$$

2.2.4 Path Length, Average Path Length and Diameter of a Network

A path in a network is defined as the set of links that connect any two nodes. For instance, $P_1 = 1 \rightarrow 3 \rightarrow 2 \rightarrow 4$ is a path between nodes 1 and 4 traversing along nodes 3 and 2 in the network shown in Fig. 2.2(a). The path length is the number of links traversed along the path so, the path length is $|P_1| = 3$ for this path. However, this is not the shortest path between nodes 1 and 4 as there is a shorter one, $P_2 = 1 \to 2 \to 4$. Thus, the shortest path length between nodes 1 and 4 is $|P_2| = 2$. We can construct the matrix of shortest path lengths of the network in Fig. 2.2(a), $\mathcal{D}_{5\times5}$,

$$\mathcal{D}_{5\times5} = \begin{bmatrix} 0 & 1 & 1 & 2 & 2 \\ 1 & 0 & 1 & 1 & 2 \\ 1 & 1 & 0 & 2 & 3 \\ 2 & 1 & 2 & 0 & 1 \\ 3 & 2 & 3 & 1 & 0 \end{bmatrix}.$$

The average shortest path length, $\overline{d}$, is the mean of the shortest path lengths across the network, given by

$$\overline{d} = \frac{1}{N(N-1)} \sum_{i \neq j}^{N} \mathcal{D}_{ij}.$$

The diameter, R, of a network is the maximum of the shortest path lengths across all pairs of nodes so, the greatest entry in matrix, $\mathcal{D}_{5\times5}$, which is equal to 3 in our example. We used the diameter of a network in the estimation of correlation decay time (CDT) from the itinerary network method in Sec. 5.2.3.

2.2.5 Clustering Coefficient

In literature, there are two types of clustering coefficients: (a) The global clustering coefficient, also known as transitivity, considers the number of connected triplets of nodes, (b) The local clustering coefficient that considers the degree of the neighbours of a node, which was introduced in [65], to quantify the small-worldness of a network. Therefore, we will compute the local clustering coefficient, C_i for node i by

$$C_i = \frac{2l_i}{k_i(k_i-1)},$$

where k_i is the number of neighbours of node i and l_i the number of links among the neighbours. Finally, we can define the mean local clustering coefficient as the average of all local clustering coefficients in the network,

$$\overline{C} = \frac{1}{N} \sum_i C_i.$$

2.3 Network Types

2.3.1 Ring Lattice

Ring lattice (RL) belongs to the family of regular network meaning that all nodes have constant degree, $deg(v_i) = k$ for all i, called k-regular network. Apart from the regular network, in a ring lattice network, all nodes are connected to $\frac{k}{2}$ neighbours on both side such as 6-ring lattice of 16 nodes, shown in Fig. 2.3(a). Each node in this network is connected to 3 neighbours on the left and 3 on the right. In its degree distribution in Fig. 2.3(b), there is only one column, as degrees of all nodes are equal to 6. The average path length and diameter of the graph increases as the network size increases, as shown in Fig. 2.3(c).

2.3.2 Random Networks

Random networks, which have been introduced by Paul Erdös and Alfred Rényi in the early 1960s, have the same probability to draw a link between any two nodes [22]. For a network with N nodes and n links, the possible number of networks can be constructed is n choose number of all links, $\frac{N(N-1)}{2}$. For example, the number of random networks can be constructed with 5 nodes and 4 links is $\binom{10}{4} = 210$ as one chooses 4 links from 10 links of a fully-connected network. Equivalently, we can assign a probability of being connected for each node, p, in the construction of random graphs where the expected value of number of edges is $\mathbb{E}(n) = p\frac{N(N-1)}{2}$. We used $p = 0.2$ to generate the network with 40 nodes in Fig. 2.4(a), consequently, we have 159 links in the network which are close to its expected value 156. The probability of being degree k in a random network with N nodes is given

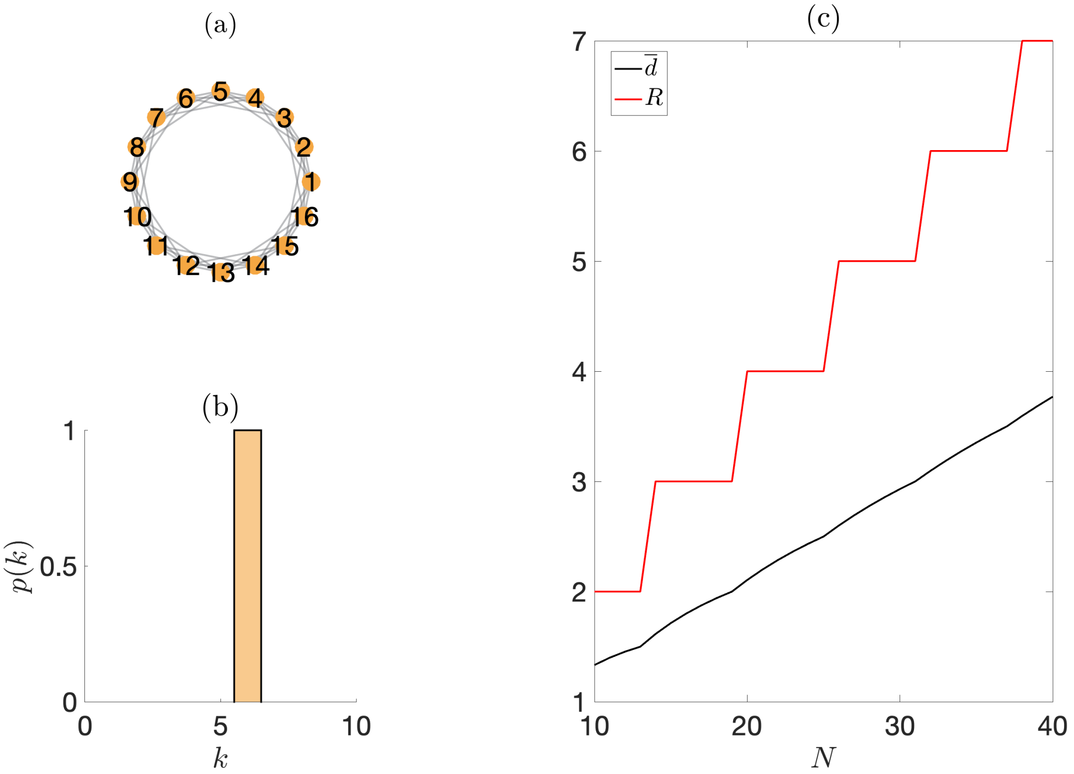

Figure 2.3: **An example of ring lattice.** (a) A network of 6-ring lattice with 16 nodes. (b) Degree distribution of the network. (c) Average shortest path length, $\overline{d}$, and diameter, R, of 6-regular lattice over network size, N.

by,

$$p(k) = \binom{N-1}{k} p^k (1-p)^{N-k-1}.$$

Any node with degree k have to be connected k nodes are chosen from the $N-1$ nodes (all nodes except itself). The probability of being connected of any two pairs is p and it implies the probability of being connected to k nodes is p^k, whereas, the probability of not being connected to the rest of the nodes is equal to $(1-p)^{N-k-1}$. Therefore, the degree distribution for the network with N nodes and connection probability p follows the binomial distribution with the parameters $N-1$ and p. It is worth noting that although its distribution becomes a Poisson distribution for large networks, the network we are working with has only 40 nodes. Therefore, we will consider that it follows a binomial distribution. We used the built-in R function, `erdos.renyi.game()`, in `igraph` package with the parameter of probability of being connected, `p.or.m`$= 0.2$. Finally, we generated

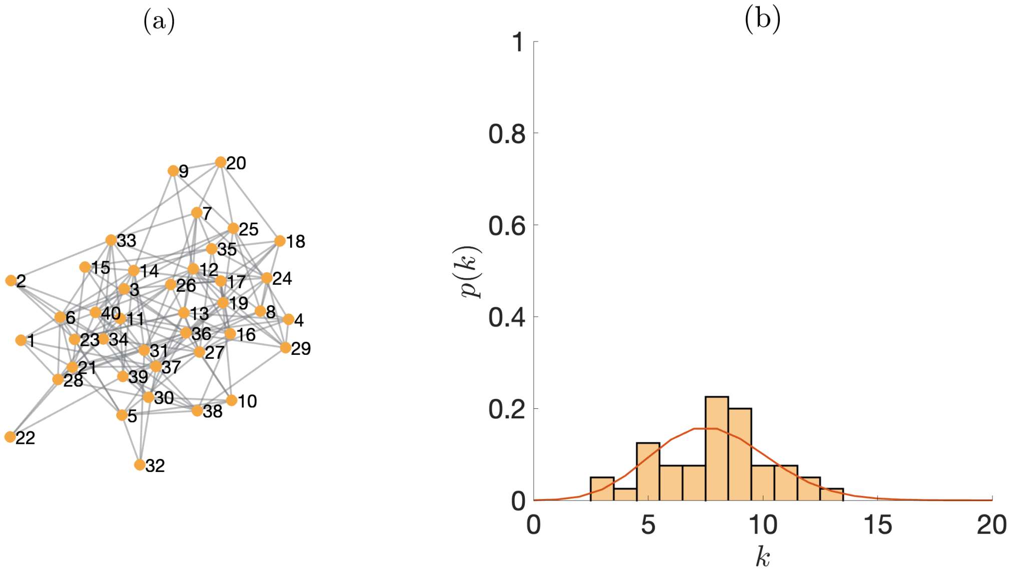

Figure 2.4: **An example of the Erdös-Rényi random network**. (a) The Erdös-Rényi random network with 40 nodes and 0.2 connection probability and (b) its node degree distribution and the curve of binomial distribution with parameters $N - 1 = 39$ and $p = 0.2$.

the ER network with 40 nodes and 159 links, as shown in Fig. 2.4(a). We presented the degree distribution of the nodes and the curve of binomial distribution, binom(39, 0.2). In a general sense, the degree distribution follows the binomial distribution, however, we can see better fit for the larger networks.

2.3.3 Small World Networks

As we see in the ring lattice, the shortest path length depends on the network size, and it increases as N rises. However, in most of the real networks, despite their often large size, they have a relatively short path length in average between any two nodes. This property is first investigated in a social science network inferred from Milgram's experiment. In his experiment, randomly selected individuals in Nebraska were asked to forward a letter to a distant target individual if they knew them, based on provided information such as names, occupation, and approximate location. If they did not know the target, they were instructed to send the letter to someone who might know the person. Milgram then tracked the path of the letters and surprisingly found that, on average, the letters reached the target individual after only 6 deliveries. It shows that a network might have relatively small mean of the shortest path lengths even if it involves many nodes. This is known as small-wordness of a network, and it helps one to characterise the network at hand. Brain networks and World Wide Webs are frequently classified as small world network because some links connects different clusters within these systems, significantly speeding up information transmission [16]. Watts and Strogatz conducted research that showed similarities in the neural networks of Caenorhabditis elegans, the power grid of the western US, and the collaboration network of film actors [65]. They developed an algorithm to generate small-world (SW) a network which has 2 parameters, number of neighbours and probability of being connected. This algorithm starts with an initial network in which all nodes connected to a given number of nearest neighbours, and the links are reconstructed based on the given probability. Since some nodes will connect to a distant node, this increase the connectivity of network and eventually provide the property of small-wordness. We used `sample_smallworld()` function in `igraph` package in R to generate Watts-Strogatz model. The parameters of the function are that dimension of the lattice, `dim`$= 1$, the number of nodes, `size` $= 40$, number of neighbours in the initial lattice, `nei` $= 4$, and rewiring probability, `p` $= 0.1$, to generate a small-world network as shown in Fig. 2.5(a) and its degree distribution is presented in Fig. 2.5(b). Small-world networks show a similar topological structure to random networks, except for the shortest path lengths.

2.3.4 Scale-Free Networks

Traditionally, ER model has been extensively used to study in complex networks, assuming an equal connection probability for all nodes. However, the analyses of large real networks show that the degree distribution of most of them follows power-law instead of binomial distribution (or Poison distribution for large networks) as shown in Fig. 2.4(b). Albert-

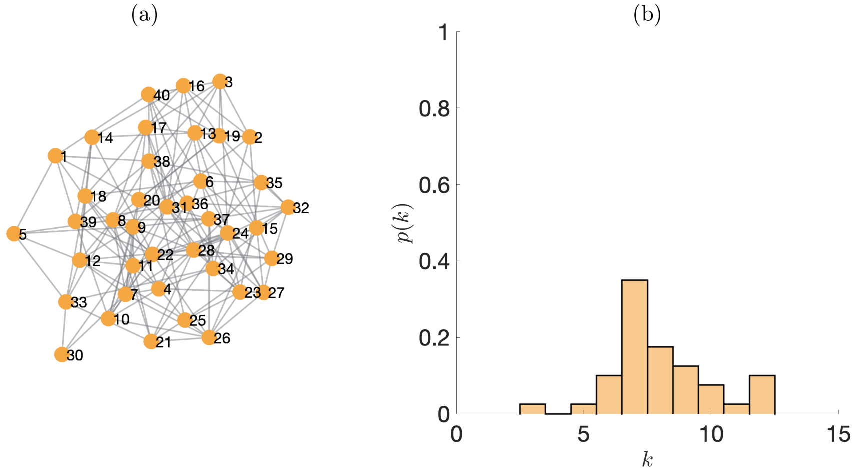

Figure 2.5: **An example of Watts-Strogatz small-world network** (a) The Watts-Strogatz small-world network with 40 nodes and 160 links and (b) its node degree distribution.

László Barabási and Réka Albert analysed the topological structure of real-world networks like the www and the citation network of scientific articles, and they showed that the nodes with higher degrees tend to have a higher connection probability, which is called “preferential attachment”. Based on this observations, they devise an algorithm to generate scale-free (SF) network, is known as BA model [5]. There are two generic mechanism underlying the algorithm: (1) New nodes get involved in the network in each iteration. (2) the probability of being connected for the nodes depends on their existing number of connections. In such networks, the probability of nodes being rewired follows a distribution with a power-law decay, denoted as $p(k) \sim k^{-\gamma}$. We generated a BA scale-free network using the built-in function in `igraph` package in R, `sample_na()`, with 40 nodes, power, $\gamma = 1$ for linear preferential attachment and `m=4`, which is the parameter for number of links will be added each step. Consequently, we obtained the network shown in Fig. 2.6(a) and its degree distribution follows the power law with $\gamma = 1$ as shown in Fig. 2.6(b).

To compare the structural properties of the networks discussed above, we presented

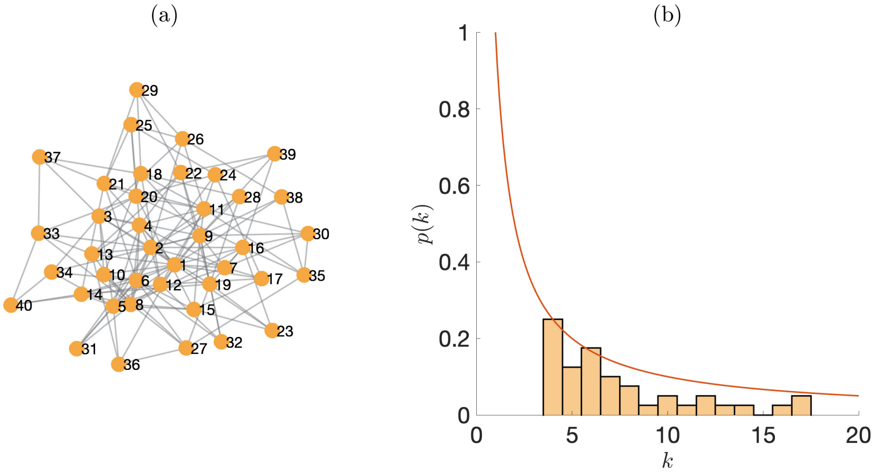

Figure 2.6: **An example of Barabási-Albert scale free network** (a) Network structure with 40 nodes and 150 links and (b) its node degree distribution, that follows the power-law with $\gamma = 1$.

some of their measures in Table 2.1. The network size and density are crucial parameters in the network inference, and we want to compare the networks on dynamics, therefore, we choose the same network size with comparable density. The diameter, the mean length of the shortest path, and the mean local clustering coefficient are the highest in RL. The average shortest path length of SW is expected to be significantly lower than that of ER, however it is slightly less because of the relatively small network size with 40 nodes. Considering the computational cost in network inference that we use these networks in Chapter 5, we kept our analysis limited to networks of 40 nodes.

	RL	**ER**	**SW**	**SF**
Number of nodes	40	40	40	40
Number of links	160	159	160	150
Density	0.21	0.20	0.21	0.19
Diameter	5	4	3	3
Average Shortest Path Length	2.95	1.97	1.94	2.00
Mean Local Clustering Coefficient	0.64	0.24	0.21	0.27

Table 2.1: Structural properties of the network topologies.

2.4 Summary

Network theory facilitates valuable analysis across various fields, from biology to the social sciences, enabling researchers to explore the interactions among the components of a system. The use of graph-theoretical tools in mathematical problems dates back to the 18th century, when Euler famously solved the “Königsberg bridge problem”, demonstrating that it is impossible to traverse all seven bridges by crossing each only once. Since then, graph theory has garnered the attention of researchers aiming to simplify and solve a range of mathematical problems.

In the 1990s, with advances in technology and computational power, network theory began to be applied to real-world problems, such as protein-protein interactions, neural connectivity in the brain, and transportation networks.

In this study, we focus primarily on undirected networks, with occasional references to weighted networks. As a result, the scope of our analysis is somewhat constrained to these structures. We also examine the structural properties of networks using concepts such as density, node degree, graph Laplacian, diameter, and clustering coefficient. Based on these structural characteristics, we introduce and analyse some of the most well-known network models, including ring lattices, random graphs, small-world networks, and scale-free networks.

Chapter 3

Brief Introduction to Nonlinear Dynamics and Chaos Theory

3.1 Nonlinear Dynamical Systems

Dynamical system theory traces its roots to Newton's contributions to science in the 17^{th} century with his book *Principia Mathematica* that has largely been accepted as one of the great revolutions in the history of physics. It leads a pioneering physical theory, so called "classical mechanics", that study the motion of objects, with three laws of motion can be explained as follows: (1) If the net force exerting on an object is zero, the object at rest remains at rest or the object in motion remains in motion with constant speed in the same direction. (2) The acceleration in a motion depends on the mass of an object and the net force applied on the object. (3) Whenever an object A exerts a force on object B, B also exerts equal and opposite force (reaction) on A. He also developed and used mathematical theories, which are the fundamentals of calculus today. His works lead to formulate the natural phenomena by mathematical equations considering infinitely small changes. Various systems from planetary positions, to air or fluid flows, to chemical reactions, can be expressed by differential equations to identify past or future states considering the infinitesimal changes in the variables of systems. In this sense, the systems whose states change over time are called *"Dynamical systems"* and their equations of motion depend on time, whether implicitly or explicitly. The theory of dynamical systems finds applications across various scientific disciplines, from biology to chemistry. For simplicity, we start with a familiar example from classical mechanics, the simple pendulum, to conceptualise the equation of motion and dynamical systems as shown in Fig. 3.1. In this frictionless (or undamped) system, the object with mass m is suspended by a bob in, with θ representing the angle between the bob and the vertical axis (or equilibrium point), called "*phase*". The position vector of the object, $\overrightarrow{s_t} = L\overrightarrow{\theta_t}$, representing the distance along the *arc* from the equilibrium axis, where L is the length of the mass-less rod.

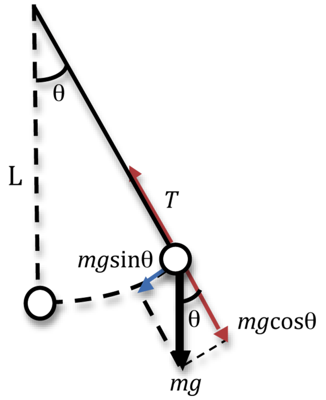

Figure 3.1: **Simple pendulum.** Forces acting on simple pendulum, where there is no friction. The figure is taken from [49].

The velocity vector, denoted by $\overrightarrow{v}$, can be defined as the rate of change of the position vector with respect to time, and the acceleration vector, denoted by $\overrightarrow{a}$, can be defined as the rate of change of the velocity vector with respect to time, given in Eqs. (3.1) and (3.2)

$$\overrightarrow{v} = \frac{ds}{dt} = L\frac{d\theta}{dt}, \tag{3.1}$$
$$\overrightarrow{a} = \frac{d^2s}{dt^2} = L\frac{d^2\theta}{dt^2}. \tag{3.2}$$

When the object is released from its current state, a force is exerted on the mass, setting it into motion. To begin, we must establish Newton's second law of motion (Eq. (3.3)), which states that the net force, $\overrightarrow{F}$, is equal to the product of mass, m, and acceleration, $\overrightarrow{a}$

$$\overrightarrow{F} = m\overrightarrow{a} \implies \tag{3.3}$$
$$-mgsin(\theta) = mL\frac{d^2\theta}{dt^2} \implies$$
$$\frac{d^2\theta}{dt^2} = -\frac{g}{L}\sin\theta. \tag{3.4}$$

where g is the gravitational acceleration. Equation (3.4) is a second-order ODE, whose

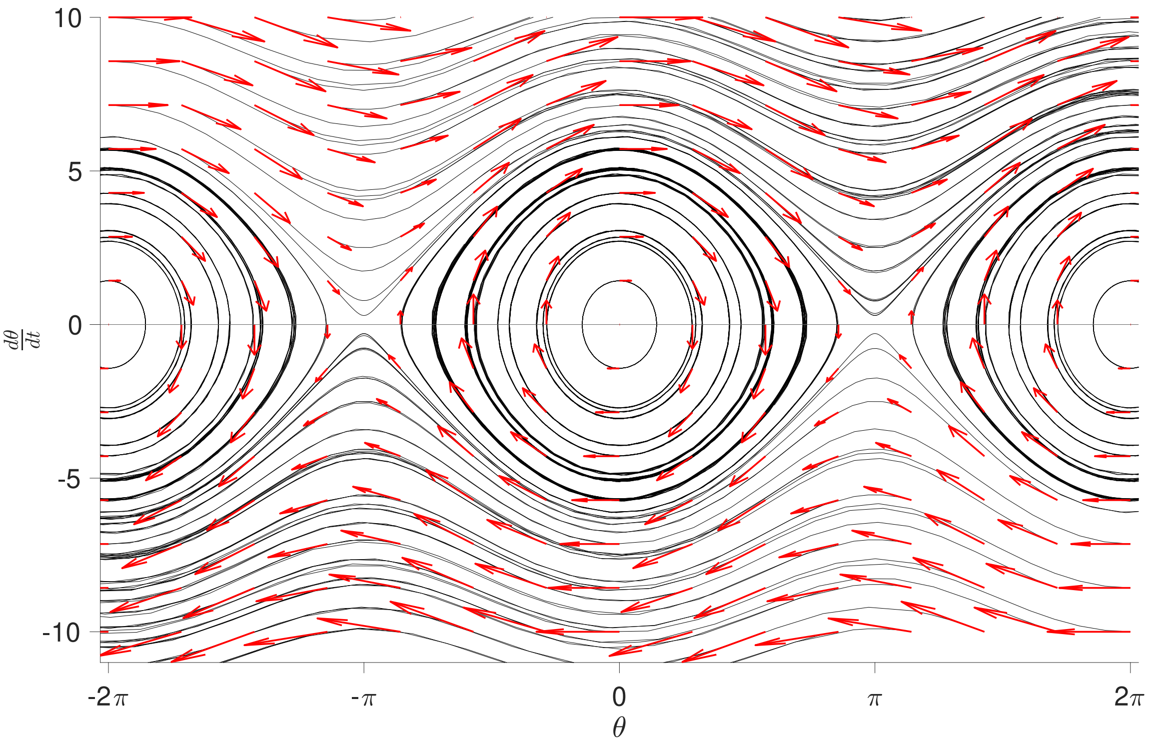

Figure 3.2: **Phase portrait of undamped simple pendulum.**

solutions determine the system. To find the solution of the system by reducing the degree of equation, we take $x = \theta$ and $y = \frac{d\theta}{dt}$. The equation becomes as follows.

$$\frac{dy}{dt} = -\frac{g}{L}\sin(x) \tag{3.5}$$

where x corresponds to the angle, θ, and y is the angular velocity, $\frac{d\theta}{dt}$. Figure 3.2 visually portrays the states of the system formulated by ODE in Eq. (3.5) for the pendulum release from various initial conditions. The trajectories with the angular velocity in $[-5, 5]$ and θ around $2k\pi$ for $k \in \mathbb{Z}$ exhibit circular motion in terms of the change in phase and angular velocity, indicating bounded oscillations. On the contrary, the trajectories centred around $\theta = 2k\pi + 1$ for $k \in \mathbb{Z}$ deviate from equilibrium by acquiring angular velocity. Due to lack of friction, the pendulum will oscillate indefinitely.

One of the main purposes of dynamical system's theory is to study the behaviour of a system. A deterministic system is a system in which no randomness is involved in its future evolution. Unlike deterministic systems, a stochastic system does not always generate the

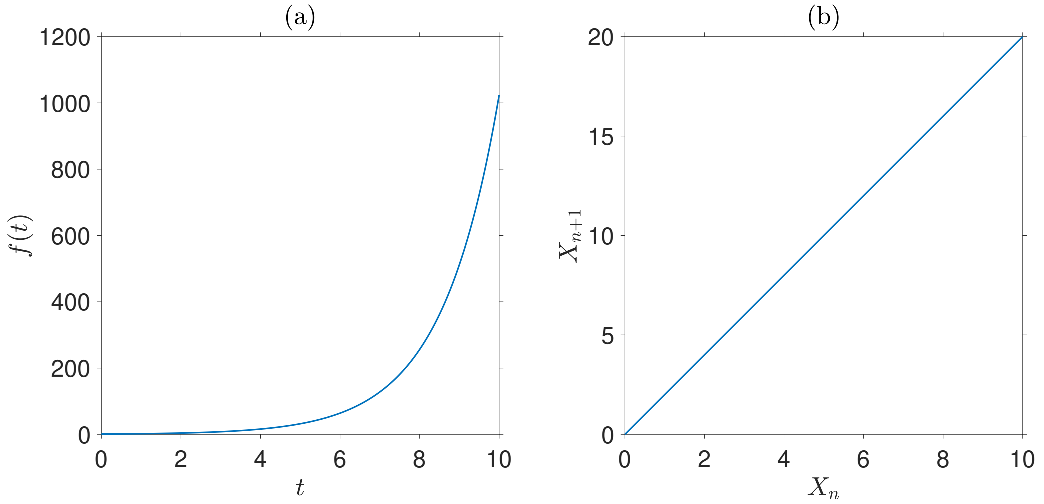

Figure 3.3: **Exponential model for population.** (a) Function of time, $f(t)$. (b) Function of past states, $f(x_n)$ (see discussion in the text on $f(t)$ and $f(x_n)$).

same output for a given input. Stochastic systems are represented by stochastic processes that arise in many contexts, e.g., in stock prices. Based on the problem at hand, time can be a ***continuous*** quantity, such as in the simple pendulum that we discussed before, or a ***discrete*** quantity, such as in the case of populations of species in ecology. In particular, if we consider an isolated island where the population of rabbits doubles every year, assuming the mortality rate 0, we can describe mathematically the population as a function of time by $f(t) = A \cdot 2^t$, where A is the initial population at time t_0. Starting with a pair of rabbits, the population will be 2 pairs of rabbits at year 1 and 4 pairs of rabbits at year 2, etc. Instead of expressing the population growth as a function of time, expressing it as a function of previous states is often preferable to see growth rate and to reduce the power order of the function. As depicted in Fig. 3.3, the population function over time exhibits an exponential trend, while the function of successive states follows a linear pattern, defined by $x_{n+1} = f(x_n) = 2x_n$, where 2 is the population growth rate.

In the context of population modelling, the issue with an exponential model is its tendency to increase indefinitely. This is because natural resources, such as food and living space, are finite and can support a finite population size. To address this, we need to add a term in $f(t)$ to constrain the indefinite growth of the population. A classical example of this idea is implemented in the Logistic map, given by

$$x_{n+1} = rx_n(1 - x_n), \tag{3.6}$$

where r is the growth rate in $[0, 4]$ and x is population in $[0, 1]$. Here, n is a number of iterations of map and plays the role of discrete time.

The logistic map in Eq. (3.6), that popularised after published as a seminal paper in 1976 by Robert May [41], holds a prominent place in dynamical systems and chaos theory due to its straightforward structure and various dynamical behaviours that depend on the growth rate r.

In this model, as the growth rate, r, falls within the interval $[0, 1]$, the population eventually decays to zero. For instance, consider an initial value of $x_0 = 0.4$ and $r = 0.2$; the future states are $0.0480, 0.0091, 0.0018, 0.0004, 0.0001, 0.0000$. This sequence of points is called as a ***trajectory*** or ***orbit***, originating from the initial condition $x_0 = 0.4$. For $r = 2$ and $x_0 = 0.4$, the trajectory will end with 0.5 called a fixed point. More broadly, a fixed point is defined as a point that remains unchanged through a trajectory, satisfying the following equation, $f(x) = x$. For $r = 2$, we have two fixed points at $x = 0$ and $x = 0.5$ which can be shown as,

$$\begin{aligned}
2x(1 - x) &= x \\
2x - 2x^2 &= x \\
2x^2 - x &= 0 \\
x(2x - 1) &= 0 \\
x &= 0 \qquad \text{or} \qquad x = 0.5.
\end{aligned}$$

Fix points can be understood as the points where the graph of a function intersects the line $y = x$, as illustrated in Fig. 3.4. Although the function gives the same value at the fixed point, nearby points can diverge or converge to a related fixed point. For example, the orbit of $x_0 = 0.4$ for $r = 2$ tends to approach $x = 0.5$, even if there is another fixed point at

$x = 0$. Even when the initial point is selected near 0, the orbit eventually converges to 0.5 after a transient period. Fixed points that attract nearby points are referred to as ***sinks*** or ***attracting fixed points***, while those that repel nearby points are known as ***sources*** or ***repelling fixed points***.

The ***Cobweb plot*** serves as a valuable tool to illustrate the behaviour of fixed points. Figure 3.4(a), (b) show the trajectory starting with the nearby point of $x = 0$ and the cobweb plot for this trajectory, respectively. After ten iterations, the trajectory reaches the sink at the fixed point $x = 0.5$.

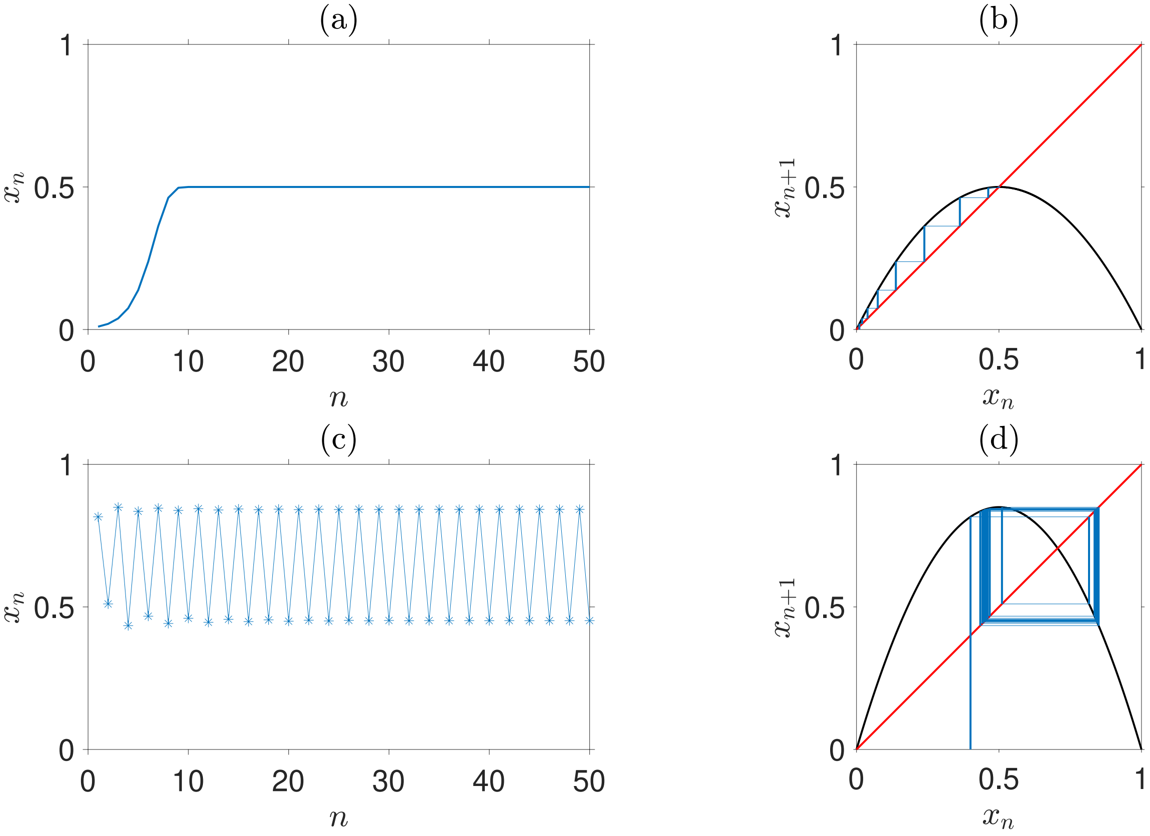

Figure 3.4: **Examples of periodic trajectories and cobweb plot** (a) An example of a trajectory of logistic map with $r = 2$ and $x_0 = 0.01$ (b) The corresponding cobweb plot, (c) An example of period 2 trajectories of logistic map with $r = 3.4$ and $x_0 = 0.4$ (d) The corresponding cobweb plot of the trajectory in panel (c).

We can analyse the stability of fixed points by examining the rate of change in ϵ-

neighbourhoods.

$$\begin{aligned} f'(0) &= \lim_{\epsilon \to 0} \frac{f(0+\epsilon) - f(0)}{\epsilon} \\ &= \lim_{\epsilon \to 0} \frac{2\epsilon(1-\epsilon)}{\epsilon} = 2. \end{aligned}$$

The distance between the nearby point $x + \epsilon$ and the fixed point $x = 0$ will double in each iteration, implying that $x = 0$ repels trajectories starting at the nearby points. However, the fixed point, $x = 0.5$, attracts nearby points as the absolute value of its rate of change is smaller than 1 as shown below.

$$\begin{aligned} f'(0.5) &= \lim_{\epsilon \to 0} \frac{f(0.5+\epsilon) - f(0.5)}{\epsilon} \\ &= \lim_{\epsilon \to 0} \frac{2(0.5+\epsilon)(1-0.5-\epsilon) - 0.5}{\epsilon} \\ &= \lim_{\epsilon \to 0} \frac{2(0.25-\epsilon^2) - 0.5}{\epsilon} \\ &= \lim_{\epsilon \to 0} \frac{-2\epsilon^2}{\epsilon} = 0 \end{aligned}$$

Then, we conclude the derivative of a function at a point gives the amount of divergence (or convergence), therefore the absolute value of the derivation at a source should be greater than 1 to repel trajectories starting at the nearby points and the absolute value of the derivation at a sink should be less than 1 to attract trajectories starting at the nearby points, as stated in the theorem [34].

Theorem 3.1.1 (Sink-source theorem). *Let f be a smooth function in $\mathbb{R}$ and p a fixed point of function f.*

1. *if $|f'(p)| < 1$, p is a sink,*
2. *if $|f'(p)| > 1$, p is a source.*

Proof. 1. Let $f'(p) = a$ and $|a| < 1$. We know $|a|$ is the contraction rate of the distance between p and $p+\epsilon$ in each iteration,

$$\begin{aligned} d(p+\epsilon, p) &= \epsilon \\ d(f(p+\epsilon), p) &= |a|\epsilon \\ &\vdots \\ d(f^k(p+\epsilon), p) &= |a|^k\epsilon = 0, \end{aligned}$$

where d denotes the Euclidean distance between two points and f^k is the k^{th} iteration of f. Since $|a| < 1$, $\lim_{k\to\infty} |a|^k\epsilon = 0$ so, p is a sink.

2. Similarly, if $f'(p) = a$ and $|a| > 1$, $\lim_{k\to\infty} |a|^k\epsilon$ go to infinity, then p is a source.

$\square$

As r increases, the logistic map demonstrates different types of dynamics, such as the behaviour of period 2 when r is set to 3.4, as shown in Fig. 3.4.(c), (d), where, following a transient period, the trajectory jumps intermittently between two points. When r is set to 3.5, the system exhibits a period 4 orbit, with the trajectory transitioning through four distinct points. Upon a slight increase in the value of r, the periodicity doubles, resulting in a period of 8 orbits, and so on. This pattern extends to period 16 orbits and continues, as illustrated in Fig. 3.5. Remarkably, there is a constant ratio in the bifurcation diagram of various chaotic maps, known as the ***Feigenbaum constant***, δ, which is equal to 4.6692. Particularly, if n^{th} period doubling (leading to period 2^n trajectories) occurs at $r = r_n$,

$$\delta = lim_{n\to\infty}\frac{r_n - r_{n-1}}{r_{n+1} - r_n} \approx 4.6692.$$

This behaviour of period doubling can be depicted by ***the bifurcation diagram***, shown in Fig. 3.5. This diagram shows the intricate long-term behaviour of the logistic map. In particular, the logistic map exhibits a diverse range of behaviours. For instance, orbits rapidly transform into chaos around $r = 3.6$ after doubling a period. An emergence of period 3 behaviour is observed around $r = 3.84$, followed by a rapid escalation of irreg-

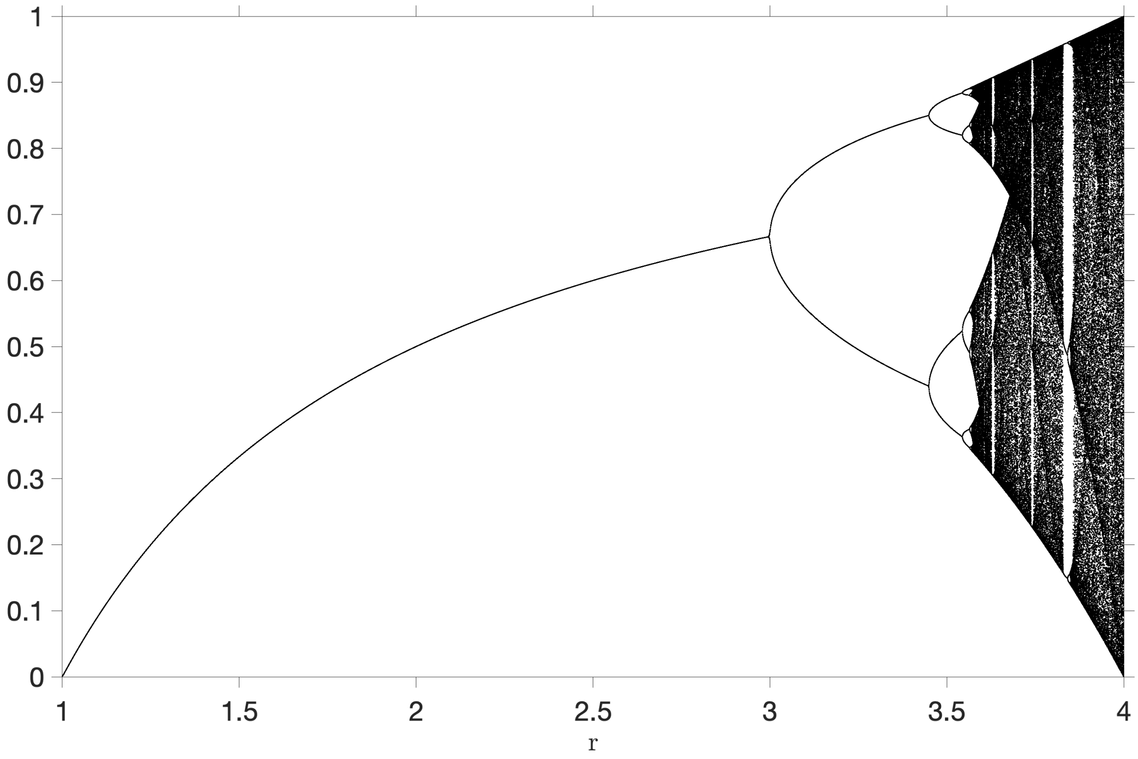

Figure 3.5: **Bifurcation diagram for logistic map.**

ularity. In Sec. 3.2, we discuss Lyapunov exponents, which is the method that quantifies the average growth (or shrinkage) of small perturbations to the solution of a dynamical system. Before that, we discuss some numerical integration methods to solve systems of ODEs next.

3.1.1 Numerical Integration Methods for ODEs

Systems of ODEs are often challenging to solve analytically; therefore, numerical integration methods are commonly used to numerically calculate their trajectories from ODE. In particular, for a continuous system (or flow), the general form of an ordinary differential equation

$$\dot{y} = f(t, y), \tag{3.7}$$

with initial condition

$$y(t_0) = y_0,$$

where $\dot{y}$ is the time derivative of y variable and $f(t, y)$ is the function of differential equation. Let the analytical solution of the system be F_t and its numerical approximation Y_t at time t. The cumulative error between the analytical solution and its numerical approximation is called **global truncation error**,

$$E(t) = F_t - Y_t.$$

The error between the analytical solution of an ODE system and its approximation in one iteration is called the **local truncation error**, and depends on the numerical method chosen. However, in practise, computers carry out the computation with finite-precision at each time step, **round-off error** arises given by

$$R(t) = y_t - Y_t,$$

where y_t is actually computed from the given numerical method. This depends on the type of computer, rounding-off method, etc.

In this section, we introduce several numerical integration techniques to approximate the solution of an ODE in Eq. (3.7), $F_t(y_0)$ presenting pseudocodes for their algorithms [19]. We implemented them to solve ODE of HR system and compared them to find the one that produces a good quality numerical solution in a reasonable amount of computational time. This comparison is based on the theory of the numerical methods presented instead of errors, since there is no simple analytical solution for HR system.

3.1.1.1 The Euler Method

The fundamental approximation technique for a solution of a system of ODEs is the **Euler Method**. To approximate further steps, it uses the first order solution of the Taylor series expansion, which is given

$$
\begin{aligned}
f(x+h) &= f(x) + hf'(x) + h^2\frac{f''(x)}{2!} + h^3\frac{f^{(3)}(x)}{3!} + \dots \\
&= f(x) + \sum_{n=1} h^n \frac{f^{(n)}(x)}{n!},
\end{aligned}
$$

where $f^{(n)}$ denotes the n^{th} derivation of function f. Using the first two terms of the Taylor series expansion, we can derive the Euler method for the solution of Eq. (3.7),

$$y_{t+1} = y_t + hf(t, y_t),$$

where h is the step size. This method computes an approximation of the function over each step using the previous approximation and the differential value of the equation at that time step. The algorithm 1 shows the pseudocode for the implementation of the Euler method.

Algorithm 1 Algorithm for Euler method

```
Define $f(t, y)$
Initial value $y = y_0$
Input step size h and number of steps n
for <j=1:n> do
    $f_n = f(t, y)$
    $y = y + hf_n$
    $t = t + h$
end for
Return t and y.
```

3.1.1.2 The 4th-order Runge-Kutta Method

The 4^{th}-order Runge-Kutta (RK4) method stands as the most widely recognised member of the Runge-Kutta numerical method family. This method approximates the analytical solution by expanding it to the 4^{th} order of its Taylor series expansion and the derivation of the coefficients derived by the author in [39]. The pseudocode for the RK4 method is given in algorithm 2.

Algorithm 2 Algorithm for RK4

```
Define $f(t, y)$
Initial value $y = y_0$
Input step size h and number of steps n
for <j=1:n> do
    $k1 = f(t, y)$
    $k2 = f(t + \frac{h}{2}, y + \frac{h}{2}k1)$
    $k3 = f(t + \frac{h}{2}, y + \frac{h}{2}k2)$
    $k4 = f(t + h, y + hk3)$
    $y = y + (\frac{h}{6})(k1 + 2 \times k2 + 2 \times k3 + k4)$
    $t = t + h$
end for
Return $t$ and $y$.
```

3.1.1.3 The 4^{th}-order Adams-Bashford Method

In contrast to the Euler and RK4 methods, the 4^{th}-order Adams-Bashford (AB4) method belongs to the category of multistep methods. This characteristic implies that the method requires values at the time steps $t - 1$, $t - 2$, $t - 3$, and $t - 4$ to estimate the value at time t. Consequently, three steps after the initial value must be calculated using one of the single-step methods, as shown in the algorithm 3.

Algorithm 3 Algorithm for AB4

```
Define $f(t, y)$
Initial values $t = t_0$ and $y = y_0$
Find the approximation of $f_1$,$f_2$, $f_3$ using a single-step method such as the Euler
  method or RK4.
Input step size h and number of steps n
for <j=1:n> do
    $y_{n+1} = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$
    $t = t + h$
end for
Return $t$ and $y$.
```

3.1.1.4 The Methods of Adaptive Runge-Kutta

The adaptive Runge-Kutta methods are based on the estimation of local truncation error, considering the methods of p and $p + 1$ order Runge-Kutta. We implemented 2^{nd}-3^{rd} order Runge-Kutta (RK23) based on the Bogacki-Shampine algorithm [17], establishing the step size by comparing the local truncation error between the RK23 with a predefined tolerance level [42], and 4^{th}-5^{th} order Runge-Kutta (RK45) based on the Dormand-Prince algorithm [21], setting the step size by comparing the local truncation error between the 4^{th} and 5^{th} order Runge-Kutta methods with a specified tolerance level [42], using two Matlab built-in functions `ode23` and `ode45` setting the relative tolerance level 10^{-8}. It should be noted that `ode23` and `ode45` were configured to compute states at equal intervals, as this aspect is crucial for the computation of the Lyapunov spectrum. Thus, the computation time of these methods increases.

3.1.1.5 Comparison of Numerical Integration Methods

In this section, our attention is directed towards the HR system in order to identify differences between the discussed numerical integration methods. The HR model replicates the activity of a single neuron, taking into account the disparity in electric potential between the interior and exterior of neurons. The molecules responsible for the membrane potential in neurons include Na^{+}, K^{+}, and Ca^{+2} ions. As a result, the model incorporates the variations in the quantities of these molecules, as depicted in the mathematical representation of the model as follows

$$\begin{aligned}\dot{p} &= q - ap^3 + bp^2 - n + I_{ext}\\ \dot{q} &= c - dp^2 - q\\ \dot{n} &= h[s(p - p_0) - n],\end{aligned}\tag{3.8}$$

with initial conditions,

$$\begin{aligned}p_0 &= -1.30784489 + 0.5\xi,\\ q_0 &= -7.32183132 + 0.5\xi,\\ n_0 &= 3.35299859 + 0.5\xi,\end{aligned}$$

where p represents the membrane potential, q is related to the fast current (Na^+ or K^+), and n is associated with the slow current (Ca^{+2}). The remaining parameters of the model are set as follows: $a = 1$, $b = 3$, $c = 1$, $d = 5$, $h = 0.05$, $s = 4$, $p_0 = -1.6$, and $I_{ext} = 3.25$. ξ is a uniformly random number generated in $[0, 1]$.

Various numerical integration methods have been used to calculate the states of the HR system. Calculations were performed for the final integration time $t_f = 10^3$, and the step sizes 10^{-4} for the Euler method, 10^{-3} for the RK4, AB4, RK23, and RK45. Here, it is worth noting that we forced the algorithms of adaptive step size, RK23 and RK45, to output at equal step size to be able to compute Lyapunov exponents later in this chapter.

Among the discussed methods, the family of Runge-Kutta integration methods is expected to be superior on Euler and Adams-Bashford because they do not consider the points between steps, therefore, we present the comparison of trajectories generated from RK4 and others (see Fig. 3.6(a)-(d)) with the cumulative absolute relative error, E_C, between trajectories (see Fig. 3.6(e)-(h)). The cumulative absolute relative error,

$$E_C = \sum_{n=1} |\frac{y_n^1 - y_n^2}{y_n^2}|,$$

where y_n^1, y_n^2 are the solutions from the method 1 and 2 in the n^{th} iteration. We used RK4 as a method 2 in all computations. The discrepancy between trajectories immediately begins to appear after time 600 in the comparison of RK4 and Euler so, the cumulative error starts to increase significantly after $t = 600$ as shown in Fig. 3.6(a), (e). The numerical solutions using the numerical integration methods, RK4, AB4, RK23 and RK45, have similar precision since no discrepancy is apparent in their trajectories. However, we can measure the difference on the microscale using E_C. The cumulative relative error of AB4 on RK4 is significantly higher than the E_C values of RK23 and RK45 in RK4. Even though the last two gives similar error functions, we know that from theory, RK45 should be the most accurate one because it considers the local truncation error between 4 and 5 order Runge-Kutta methods. The computation time of the integration methods is presented in Table 3.1. Even if RK4 is the most efficient method in terms of the accuracy of the

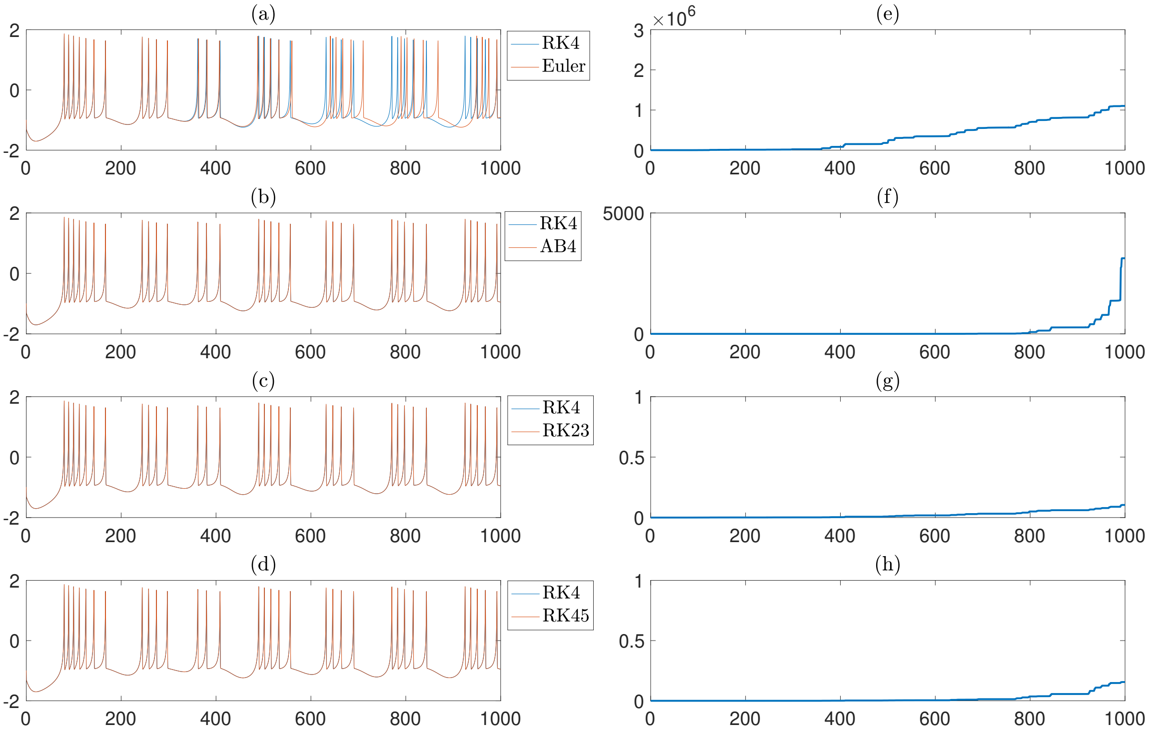

Figure 3.6: **Comparison of numerical integration methods.** (a)-(d) Comparison of the trajectories of p variable of HR system using various numerical integration methods. (e)-(h) The corresponding cumulative relative error, E_C, of the methods on RK4 as a function of time.

computed trajectories and computational time, we use RK45, unless otherwise stated, because the chaotic orbits are highly sensitive to initial conditions and we prefer to use the most accurate method possible.

Method	**CPU time** (in seconds)
Euler	67
RK4	24
AB4	22
RK23[1]	144
RK45[1]	219

[1] The method is adapted to output in constant step size.

Table 3.1: CPU time of numerical integration methods (in seconds).

3.2 Chaos Theory

As discussed previously, the logistic map demonstrates diverse and intricate dynamical behaviours at various values of the parameter r. In summary, for $r \in (0, 1)$, the trajectories of the logistic map decay to zero. For $r \in (1, 3)$, the trajectories converge to a fixed point. Beyond this point, as r increases from 3, one observes the emergence of the regular behaviour including period 2, period 4, period 8, etc. Chaotic behaviour ends up occurring around $r = 3.6$. A fascinating pattern emerges where the dynamical behaviours transition from regular to chaotic and back to regular. For a brief interval around $r = 3.82$, the dynamics exhibit regularity with period 3, period 6, and so forth. However, this regular behaviour is not permanent, and the system quickly returns to chaotic behaviour. The primary objective of this section is to delve into the concept of the Lyapunov exponent, which quantifies the degree of irregularity, to provide a precise definition of chaos.

3.2.1 Lyapunov Exponents for 1-dimensional maps

As outlined in Sec. 3.1, the stability of fixed points hinges on the derivative of the function at those fixed points p. We showed that p acts as a source when $|f'(p)| > 1$, and, conversely, it functions as a sink when $|f'(p)| < 1$. We also discussed that the derivation of the map at a certain point gives the growth (or shrinkage) rate for the nearby points. Lyapunov exponents is a measure to quantify the average growth (or shrinkage) rate along the trajectory. The Lyapunov exponent for an 1-dimensional map f with its trajectory $x_t = x_0, x_1, x_2, x_3, ...$ can be derived as follows

$$h(x_1) = \lim_{k \to \infty} [f'(x_1) f'(x_2) \ldots f'(x_k)]^{1/k},$$

where $h(x_1)$ is called the *Lyapunov number* which gives the average growth (or shrinkage) rate and *Lyapunov exponent* equals to its logarithm, $\log h(x_1)$,

$$\begin{aligned}\lambda(x_1) &= \lim_{k\to\infty} \frac{\log|f'(x_1)f'(x_2)....f'(x_k)|}{k} \\ &= \lim_{k\to\infty} \frac{\log|f'(x_1)f'(x_2)....f'(x_k)|}{k} \\ &= \lim_{k\to\infty} \frac{\log|f'(x_1)| + \log|f'(x_2)| + ... + \log|f'(x_k)|}{k}.\end{aligned}$$

3.2.2 Lyapunov Exponents for Multi-dimensional Systems

In this section, our exploration begins with a study of 2-dimensional linear transformations to clarify the concept of expansion rate. Subsequently, our focus shifts towards 2-dimensional non-linear transformations. Finally, we consolidate our discussion by elucidating the concept of the Lyapunov exponent and providing a precise definition of Chaos.

3.2.2.1 Matrix Transformation

Let's start with a linear transformation of $f(x, y) = (x+2y, 2y+x)$, which can be expressed by a transformation matrix, T=$\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$. T has 2 eigenvalues: $\lambda_1 = 3$ associated with the eigenvector $(1, 1)$ and $\lambda_2 = -1$ associated with the eigenvector $(-1, 1)$. In Fig. 3.7(a), the blue circle shows the initial circular region in a 2-dimensional space with $r = 0.1$ centred at the origin. The 1^{st} iteration of the region under the given linear transformation is shown with a red ellipse. The blue circle expands through its eigenvector $(1, 1)$, as its associate eigenvalue $\lambda_1 = 3$ is greater than 1. Since $\lambda_2 = -1$ and its absolute value equals 1, ellipse maintains its size along the eigenvector $(-1, 1)$. However, the circle transforms to the different shape under the nonlinear transformation shown in Fig. 3.7(b). To produce these results, the function $f(x, y) = (x+2xy, 2x+y)$ is used, which is a nonlinear transformation because of the term $2xy$. Its transformation matrix can be written as T=$\begin{pmatrix} 1 & 2x \\ 2 & 1 \end{pmatrix}$ that depends on x unlike the linear transformation, its eigenvalues and eigenvectors will change as a function of x. It means that the direction of expansion and the expansion rate change according to the position of the point, and this implies that the transformation of the circle under the nonlinear function is not an ellipse any more. However, when the small circle is initially chosen, the transformed region is close to an ellipse, as seen in Fig. 3.7(c).

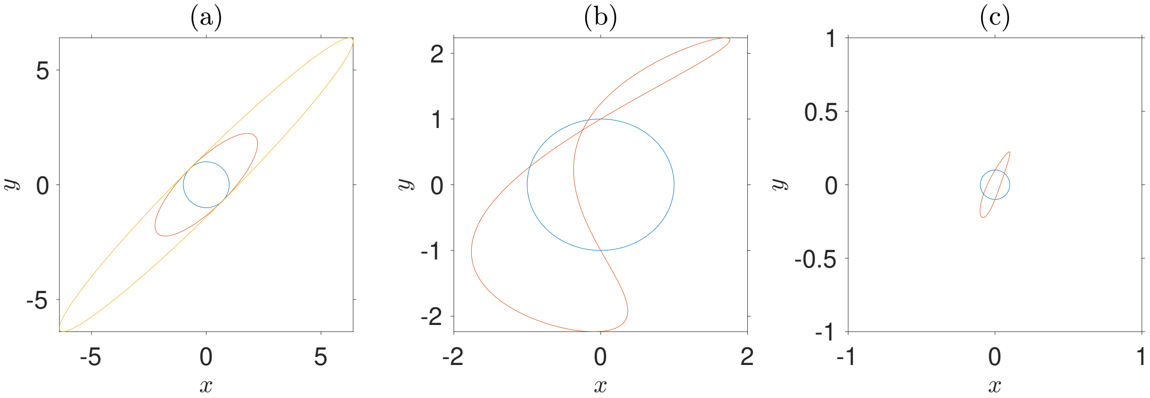

Figure 3.7: **Linear and nonlinear transformation**. (a) Linear Transformation. Blue circle is initial circular region $r = 0.1$ at $(0, 0)$ centred, red circle is 1^{st} iteration under f transformation, orange circle is 2^{nd} iteration under f transformation. (b) Nonlinear Transformation of the same circle. (c) Nonlinear Transformation of the small size circular region with $r = 0.01$.

3.2.2.2 Lyapunov Exponents

The idea behind measuring the average expansion (or shrinkage) rate of an orbit under a nonlinear transformation is to make small perturbations along the trajectory. Only in this way does the nonlinear transformation act as a linear one and makes it possible to predict the average expansion. In this subsection, we dive into the details of the algorithm to compute Lyapunov exponents in multidimensional systems, presented in [7,8].

The idea behind the computation of the Lyapunov exponents involves the evolution of the Lyapunov vectors, initially chosen as unit basis vectors, shown in Fig. 3.8(a), (b) under the Jacobian matrix, $\mathbf{J} = Df$, where D is the partial derivation operator along all variables. Following evolution of the Lyapunov vectors, the Gram-Schmidt orthogonalisation process is used to ensure that the Lyapunov vectors are separated enough to not cause any computational errors, as shown in Fig. 3.8(c). The length of these orthogonal vectors can be interpreted as the expansion rate in this iteration. To approximate the Lyapunov exponents, the average of the logarithms of the lengths of these deviation vectors is calculated as a function of n. It is important to note that theoretically, n goes to infinity. As

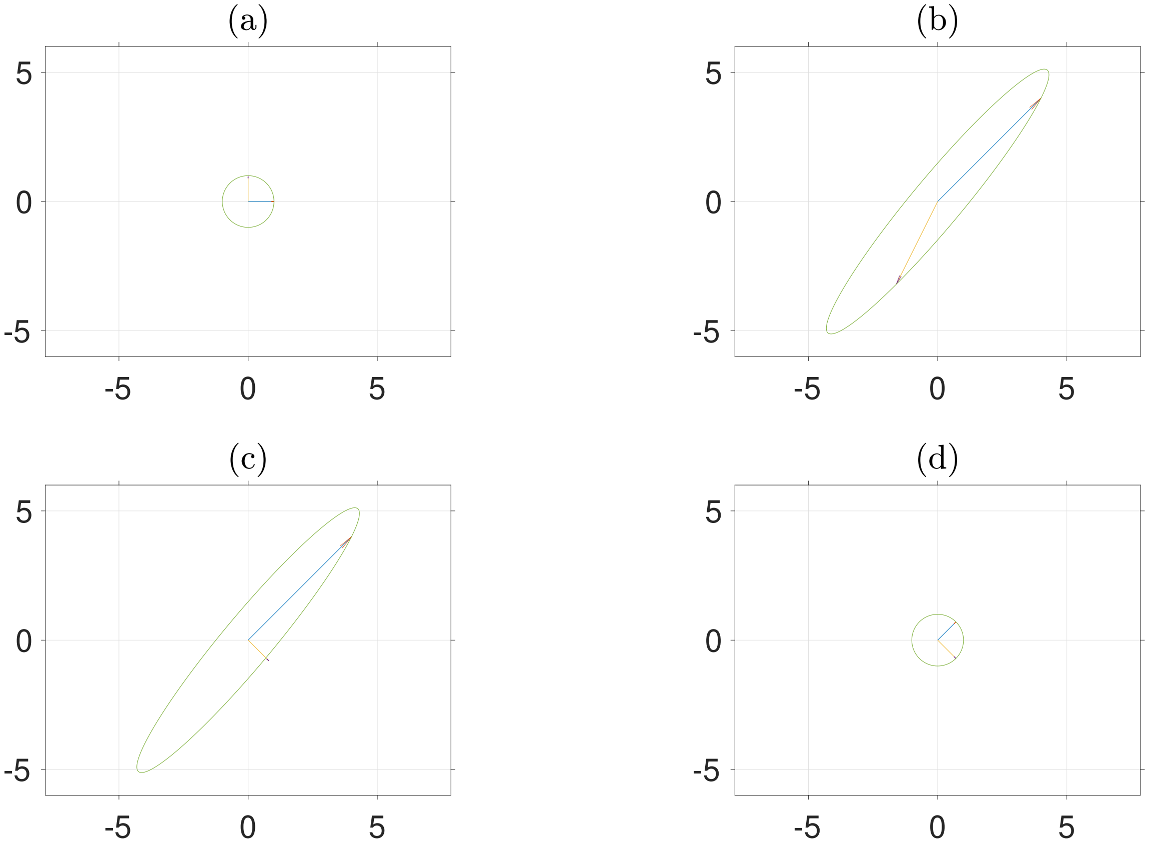

Figure 3.8: **Evolution of Lyapunov vectors**. (a) Initial Lyapunov vectors (b) Evolution of Lyapunov vectors (c) Orthogonalisation of the vectors used in the Gram-Schmidt method (d) Normalisation of the Lyapunov vectors to be used again in the next iteration

shown in Fig. 3.8(d), it is necessary to normalise the orthogonal vectors before proceeding to the next iteration. This normalisation step plays a crucial role in maintaining control over the lengths of the vectors, thus effectively mitigating over expansion.

Definition 3.2.1 (Lyapunov exponents of maps)**.** *Let f be a smooth map on R^M, let $\mathbf{J}_n = Df_n(x_0)$, r_k^n be the length of the k^{th} longest orthogonal axis of the ellipsoid $\mathbf{J}_n N$ for an orbit with initial point x_0 and for $k = 1, ..., M$, n is the length of time series. Then r_k^n measures the contraction or expansion near the orbit of x_0 during the first n iterations. The k^{th} Lyapunov number of x_0,*

$$h_k = \lim_{n\to\infty} (r_k^n)^{\frac{1}{n}}, \tag{3.9}$$

if this limit exists. The k^{th} *Lyapunov exponent of* x_0 *is* $\lambda_k = \log h_k$*. Notice that we have built into the definition the property that* $h_1 \geq h_2 \geq \ldots \geq h_M$ *and* $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M$*.*

For continuous systems (or flows), the concept of Lyapunov exponents is the same once we replace the iteration of discrete map with a differential equation of a flow. Let $F_t(y)$ be a solution of the Eq. (3.7) so, it should satisfy the equation.

$$\frac{d}{dt}F_t(y) = f(F_t(y))$$

Once we take the derivative of both sides with respect to y, we get the so-called variation equation.

$$\frac{d}{dt}DF_t(y) = Df(F_t(y))DF_t(y) \tag{3.10}$$

In order to simplify the Eq. 3.10, we can rename $\mathbf{J_t}(y) = DF_t(y)$ and $\mathbf{\Lambda(t)} = Df(F_t(y))$. Then,

$$\dot{\mathbf{J}}_t = \mathbf{\Lambda(t)J_t}. \tag{3.11}$$

To solve Eq. (3.11), we take $\mathbf{J_0} = \mathbf{I}$ which corresponds to the unit vectors shown in Fig. 3.8. For the 3-dimensional HR system introduced in Eq. (3.8),

$$\mathbf{\Lambda(t)} = \begin{bmatrix} -3ap^2(t) + 2bp(t) & 1 & -1 \\ -2dp(t) & -1 & 0 \\ hs & 0 & -h \end{bmatrix}.$$

Notice that $\mathbf{\Lambda(t)}$ involves the p variable at time t hence, the ODE of a system and its variation equation in Eq. (3.11) should be solved simultaneously. To control the lengths of the column vectors of $\mathbf{J}$, we used Gram-Schmidt orthonormalization and recorded the length of the vectors of $\mathbf{J}$ at each step, following the same approach in Fig. 3.8.

Definition 3.2.2 (Lyapunov exponents of flows)**.** *Let* $\dot{y} = f(t, y)$ *be a differential equation of a continuous time system,* $\dot{\mathbf{J}} = \mathbf{\Lambda J}$ *variation equation where* $\mathbf{\Lambda}$ *is the Jacobian matrix of the differential equation with size* $M \times M$*. Let* r_k^n *be the length of the* k^{th} *longest orthogonal variation vector of* J *for an orbit with initial point* y_0*, for* $k = 1, ..., M$*. Then* r_k^n *measures* *the contraction or expansion near the orbit of* y_0 *during the first* n *iterations. The* k^{th} *Lyapunov number of* y_0*,*

$$h_k = \lim_{n\to\infty} (r_k^n)^{\frac{1}{n}},$$

if this limit exists. The k^{th} *Lyapunov exponent of* y_0 *is* $\lambda_k = \log h_k$*. Notice that we have built into the definition the property that* $h_1 \geq h_2 \geq \ldots \geq h_M$ *and* $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M$*.*

After calculating the Lyapunov spectrum, which is the set of Lyapunov exponents, we can determine whether the system is chaotic. This relies on the following definitions of chaos for maps and flows stated in [34],

Definition 3.2.3 (Chaotic map)**.** *Let* f *be a map of* $\mathbb{R}^m$*,* $m \geq 1$*, and let* $x_t = (x_0, x_1, x_2, \ldots)$ *be a bounded orbit of* f*. The orbit is chaotic if the following conditions are satisfied.*

1. *It is not asymptotically periodic,*
2. *No Lyapunov Exponent is exactly zero,*
3. *There is at least one positive Lyapunov Exponent.*

In this definition, we exclude quasiperiodic trajectories with the 2^{nd} condition due to their high predictability. The authors in [34] present the following example to elucidate quasi-periodic behaviour.

Example: Let f be a function that iterates the given point in polar coordinates, defined by the following equation

$$f(r, \theta) = (r^2, \theta + \psi),$$

where r is the distance to the origin and θ the angle. For points $r < 1$, the trajectories approach the fixed point at the origin. For points $r > 1$, the region is unbounded and the trajectories tend toward infinity. Quasi-periodicity arises when $r = 1$ and ψ is an irrational number. The trajectories exhibit almost periodic behaviour, while being sensitive to initial conditions at the same time. The Lyapunov exponents of the trajectories are $\log 2$ and 0.

Definition 3.2.4 (Chaotic flow)**.** *Let* $F_t(y_0)$ *be a solution of* $\dot{y} = f(t, y)$ *where* y_0 *in* $\mathbb{R}^N$*.* $F_t(y_0)$ *is chaotic orbit if the following conditions hold:*

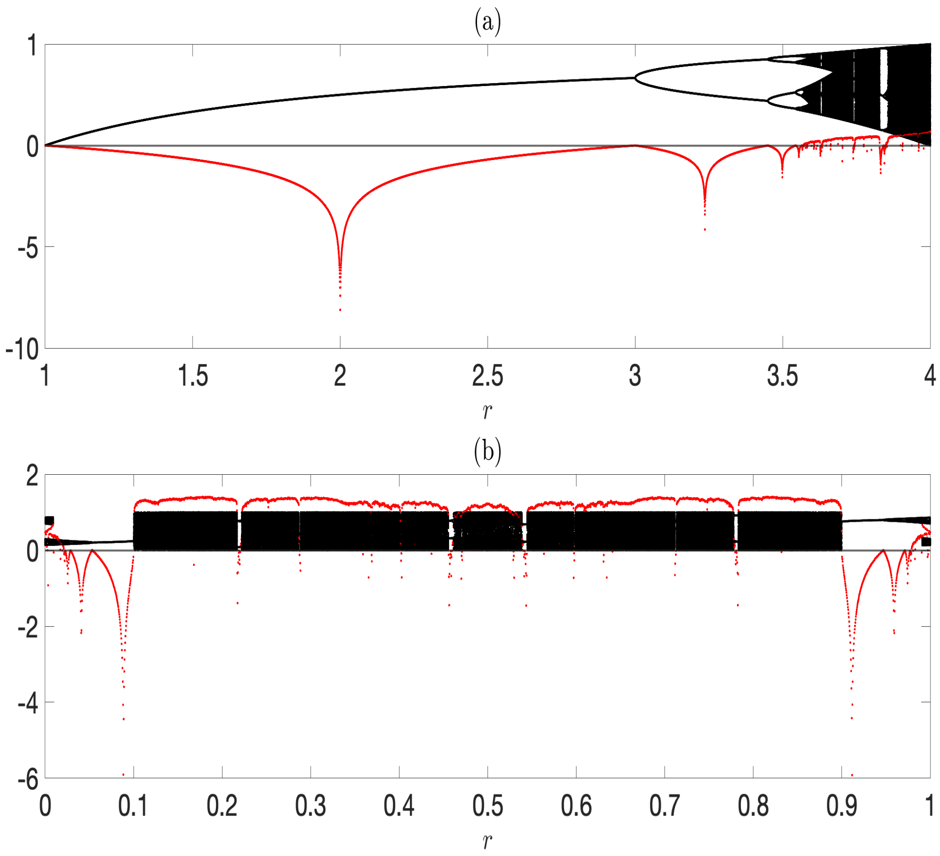

Figure 3.9: **Bifurcation diagram and MLE**. Plot of the bifurcation diagram and MLE as a function of r for the logistic and circle maps. The red curves show MLEs as a function of r, and the black curves are the corresponding bifurcation diagrams of the logistic map in panel (a) and the circle map in panel (b).

1. *There is at least one positive maximum Lyapunov exponent (MLE). This implies that the system is sensitive to the initial condition.*
2. $F_t(y_0)$ *is bounded.*
3. *The long-term behaviour of the trajectory is not periodic.*

We examined previously the dynamics of the logistic map depending on r and provided the bifurcation graph. In Fig. 3.9(a), the bifurcation graph of the logistic map and MLE is presented together. It is apparent that the MLE is negative, where the trajectory is nonchaotic. Similarly, we plot the bifurcation diagram and corresponding MLEs in for the

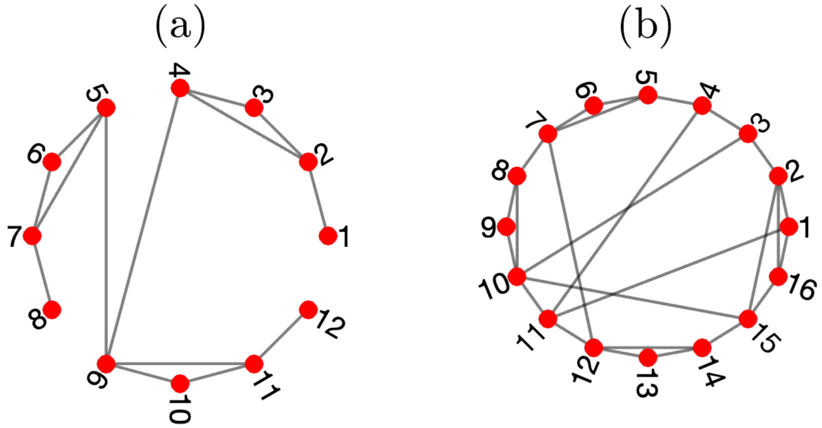

Figure 3.10: **Simple network topologies.** (a) A network of 12 nodes and 14 links. (b) Network of 16 nodes and 26 links.

circle map in Fig. 3.9(b) is given by,

$$f(x,r) = x + r - \frac{K}{2\pi}\sin(2\pi x_n) \qquad (\text{mod } 1), \tag{3.12}$$

where r is a parameter in $[0,1]$, x the input variable in $[0,1]$ and $K = 6.9115$. Figure 3.9(b) reveals various dynamics similar to that observed in the logistic map. For the system to be chaotic, we have chosen $r = 0.35$.

In this thesis, we study the interaction among the units of a system whose data are generated by coupled discrete and continuous dynamical systems, discussed next.

3.3 Coupled Dynamical Systems

We employ coupled discrete and continuous deterministic systems, as well as the Kuramoto system augmented with a stochastic term. In this chapter, we used two network topologies, one consisting of 12 nodes in Fig. 3.10(a) and another with 16 nodes in Fig. 3.10(b) to generate data from coupled models.

Index	**Logistic map**	**Circle map**	**Index**	**Logistic map**	**Circle map**
1	0.45	1.21	**9**	0.39	0.96
2	0.44	1.18	**10**	0.38	0.93
3	0.44	1.15	**11**	0.37	0.88
4	0.43	1.12	**12**	0.35	0.83
5	0.42	1.10	**13**	0.34	0.77
6	0.42	1.07	**14**	0.31	0.70
7	0.41	1.03	**15**	0.29	0.62
8	0.40	1.00	**16**	0.24	0.47

Table 3.2: Lyapunov spectra of 16 coupled logistic and circle maps for coupling strength $\alpha = 0.1$.

3.3.1 Deterministic Discrete Systems

3.3.1.1 The Coupled Logistic and Circle Maps

The general equation used to generate trajectories from coupled logistic and circle maps is given by

$$x^i_{n+1} = f(x^i_n, r)(1-\alpha) + \frac{\alpha}{k_i} \sum_{j=1}^{M} \mathbf{A}_{ij} f(x^j_n, r). \tag{3.13}$$

Here, x^i_n represents the n^{th} iteration of the i^{th} node, M is the number of nodes in the network whose adjacency matrix $\mathbf{A}_{ij}$, depicted in Fig. 3.10(b). The parameter α stands for the coupling strength and k_i is the total degree of the i^{th} node. The function $f(x^i_n, r)$ is the map that defines the dynamics on the i^{th} node, can be the logistic or circle maps, given in Eqs. (3.6) and (3.12), respectively. The initial conditions are randomly chosen in $[0, 1]$.

The coupling strength is one of the factors that affects the dynamics of the system. In Table 3.2, the Lyapunov spectra of 16 coupled logistic and circle maps in Eqs. (3.3), (3.6) and (3.12), where the coupling strength $\alpha = 0.1$, is presented. We see that both Lyapunov spectra are positive, indicating that the dynamics is chaotic in both cases.

3.3.2 Deterministic Continuous Models

3.3.2.1 The Coupled Hindmarsh-Rose System

In addition to the HR system introduced in Eq. (3.8), we use here, a coupling term to generate trajectories based on interactions among nodes by

$$\begin{aligned} \dot{p_i} &= q_i - ap_i^3 + bp_i^2 - n_i + I_{ext} - g_l \sum_{j=1}^{M} \mathbf{C}_{ij} H(p_j) \\ \dot{q_i} &= c - dp_i^2 - q_i \\ \dot{n_i} &= h[s(p_i - p_0 - n_i) \qquad \text{for} \quad i = 1, 2, ..., M., \end{aligned} \tag{3.14}$$

where M is the number of neurons, $H(p_j) = p_j$, and $\mathbf{C}_{ij}$ is the Laplacian derived from equation $\mathbf{C} = \mathbf{A} - \mathbf{D}$, where $\mathbf{A}$ is the adjacency matrix of the network and D the degree matrix of A, with its diagonal indicating the degrees of the associated nodes (see more details in Chapter 2). Finally, g_l denotes the coupling strength of the electrical connections and all other parameters are set as in Eq. (3.8). The initial conditions are chosen as follows [12]

$$\begin{aligned} p_0^i &= -1.30784489 + 0.5\xi^i, \\ q_0^i &= -7.32183132 + 0.5\xi^i, \\ n_0^i &= 3.35299859 + 0.5\xi^i, \end{aligned}$$

where ξ^i is a uniformly distributed random numbers of node i. In Sec. 3.1.1, we explored the methods employed to numerically solve the HR system. Based on the interactions in the network of 12 nodes seen in Fig. 3.10(a), we have generated a trajectory (data set) with the final integration time $t_f = 10^4$ and step size $h = 0.1$ for the single and coupled models.

The algorithm forces `ode45` (built-in Matlab function) to produce data with a time step, $h = 0.1$ as it is an adaptive step-size numerical integration method. We start recording the orbit after the MLE stabilises. To do so, we compute the MLE as a function of time and

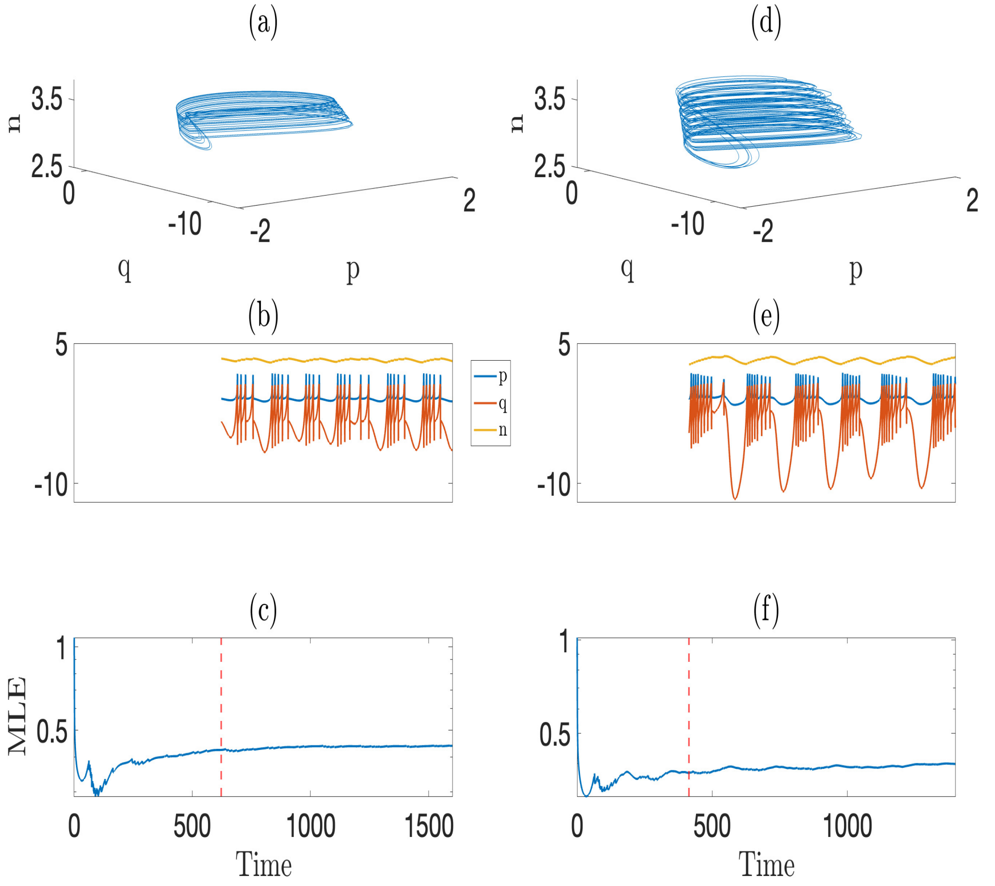

Figure 3.11: **The dynamics of single and coupled HR systems.** Panels (a) and (d) show the trajectories of node 1 for uncoupled and 0.1 coupled HR model are respectively shown in 3d space. Panels (b) and (e) show the recorded time series of variables p, q, n of node 1 for the models. Panels (c) and (f) show the time evolution of MLE has shown in log scaled y-axes for the corresponding models. The data were recorded after MLEs got stabilised, indicating by red dashed line.

compute the standard deviation of a sliding time window of 500 MLEs. When the standard deviation of 500 MLEs in a window is smaller than a predefined threshold, we consider that the MLE has converged to a value. If this value is positive, it is an indication that the dynamics is chaotic, and we start recording the time series after the last of these 500 time points. In this context, we choose the predefined threshold as 10^{-3} and start recording time series after the time point indicated by the vertical red-dash line in the panels (c) and (f) in Fig. 3.11. In panels (a), (d), we plot two trajectories of node 1 in the 3-dimensional space for uncoupled and coupled systems, respectively. Similarly, panels (b) and (e) show the p, q and n variables as a function of time. Finally, panels (c) and (f) show the evolution of MLEs on logarithmic scale for the uncoupled and coupled systems that approach 0.442 and 0.402, respectively, as time increases.

3.3.2.2 The Coupled Lorenz System

With significant advancements in computational power after the 1970s, numerical integration methods have facilitated the solution of ordinary differential equations. This has led to good estimations of the future states of a system of ODEs. Consequently, scientists have been able to determine precise dates for solar and lunar eclipses thousands of years ago or in the distant future. The application of these methods culminated in the historic landing of “Apollo 11” on the Moon in 1969.

This progress was driven by the ability of scientists to formulate equations of motion that encompassed all relevant factors influencing a particular process. During the 1960s, climate scientists believed that with a comprehensive equation of motion accounting for all variables affecting weather conditions, they would achieve absolute certainty in weather forecasts. The belief was that weather prediction could be extended over long time spans, similar to predicting the movement of celestial bodies.

However, the landscape of scientific thought shifted with the work of Edward Norton Lorenz in 1963, a mathematician and meteorologist at the Massachusetts Institute of Technology. Lorenz is widely recognised as the pioneer of **“Chaos Theory”** due to his groundbreaking discovery. While he was studying the atmospheric model, included 12 differential equations and many parameters that he does not know how to set. The use of a computer helps him to investigate phase space with various parameters. When he computed the numerical solution of the system for the same initial condition, Lorenz noticed that it gives quite different solutions after some iterations [25]. Before Lorenz, the common sense was that if an equation of a deterministic system could be written, it is predictable. However, it has been destroyed by Lorenz’s finding of the initial sensitive system. Years later, he derived the three-dimensional system of ordinary differential equations (ODE) modelling the convection between two plates parallel to the ground: one cooling uniformly from above, and the other heating uniformly from below, given by

$$\begin{aligned} \dot{x_i} &= \sigma(y_i - x_i) \\ \dot{y_i} &= x_i(\rho - z_i) - y_i \\ \dot{z_i} &= x_i y_i - \beta z_i, \end{aligned} \tag{3.15}$$

where the model parameters are $\sigma = 10$, $\rho = 28$, $\beta = 8/3$ that leads to chaotic orbit. The variable x is proportional to the speed of circulatory convection, y to the horizontal heat variation and z to the vertical heat variation.

In the discussion of local stability of a logistic map, we define the concept of sink as an attracting fixed point, meaning that the points in its basin eventually reach the sink. In more general sense, in the dynamics of Lorenz system, we see a butterfly-shaped curve in three dimensions that attracts trajectories of many starting points. In other words, the attractor is a closed set, A, of its phase space, which trajectories of many starting points evolve to A. Because of the chaotic trajectories attract to this butterfly-shaped close-set, Lorenz system is also called “chaotic (or strange) attractor”. It is worth noting that this discussion is valid for only the specified parameters, as Lorenz system has different dynamics based on the parameters (see [34] for the discussion of the dynamics of Lorenz system based on the parameter ρ).

In our experiments, we only use the x component coupled Lorenz system in Eq. (3.15) as the coupling function is present only in the first equation in Eq. (3.16)

$$\begin{aligned} \dot{x_i} &= \sigma(y_i - x_i) + K \sum_{j=1}^{M} \mathbf{A}_{i,j}(x_j - x_i), \\ \dot{y_i} &= x_i(\rho - z_i) - y_i, \\ \dot{z_i} &= x_i y_i - \beta z_i. \end{aligned} \tag{3.16}$$

where K is the coupling strength, A the adjacency matrix of M connected nodes. The initial conditions of the system are $x_0^i = 0.1 + 0.5\xi^i$, $y_0^i = -0.19 + 0.5\xi^i$ and $z_0^i = -0.27 + 0.5\xi^i$,

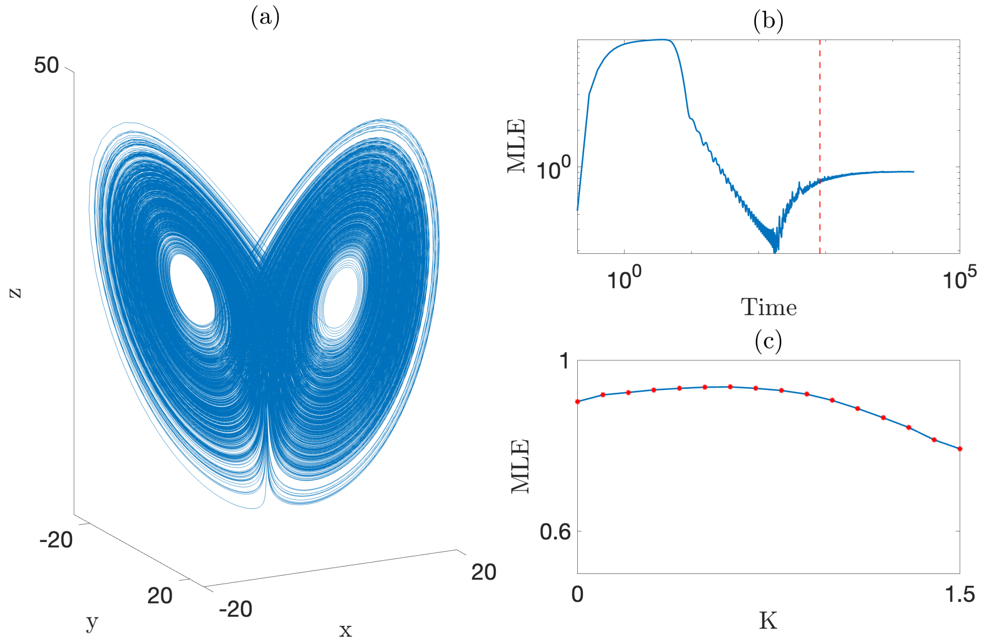

Figure 3.12: **Dynamics of the single and coupled Lorenz Systems**. (a) The Lorenz attractor of system (3.15) (single Lorenz system). (b) The time evolution of the MLE which approaches to 0.9 as time increases (for coupled Lorenz system for $K = 0.6$ in Eq. (3.16)) (c) Change in the MLE over the coupling strength of coupled system (3.16), notice that the curve peaks at 0.94 for $K = 0.6$.

where ξ^i is a uniform random numbers in $[0, 1]$ for $i = 1, \ldots, M$. The system is numerically solved using the RK45 numerical integration method, which uses internally a variable time step to keep the local truncation error below a small threshold (we used 10^{-8} in our simulation). Since we want the numerical solution at specific time intervals, we “force” the method (RK45) to report the values at multiples of the time step 0.1 and use as final integration time, $t_f = 2 \times 10^3$. Figure 3.12(a) presents the trajectory of the uncoupled Lorenz model in 3D space, where we can see the butterfly-like shape of the so-called Lorenz attractor. Figure 3.12(b) shows the evolution of the MLE of uncoupled model over time in log scaled y-axis, which approaches to 0.9029 as time goes to infinity. We start recording the orbit after the MLE stabilises. To do so, we compute the MLE as a function of time and compute the standard deviation of a sliding time window of 500 MLEs. When the standard deviation of the 500 MLEs in a window is smaller than a predefined threshold, we consider that the MLE has converged to a value. If this value is positive, it is an indication the dynamics is chaotic, and we start recording the time series after the last of these 500 time points. In this context, the trajectory is recorded once the MLE stabilises, under the assumption of that the standard deviation of 500 MLEs within sliding time windows is less than or equal 10^{-2}. Thus, first 1935 data points are considered as transient period indicated by vertical red-dashed line. Figure 3.12(c) illustrates the MLE of the system for K values ranging from 0 to 1.5. The MLE reaches its peak value of approximately 0.94 around $K = 0.6$.

3.3.2.3 Deterministic Coupled Kuramoto Oscillators

The phenomenon of synchrony among fireflies in Malaysia, that they flash in unison, has intrigued travellers for centuries, often described in romantic terms in the travel literature (see details in Sec. 4.1). The Kuramoto model has contributed significantly to the understanding of this phenomenon by the following equation [37],

$$\dot{\theta}_i = \omega_i + \frac{K}{M} \sum_{j=1}^{M} \mathbf{A}_{i,j} \sin(\theta_i - \theta_j), \tag{3.17}$$

where θ_i is oscillator phases for $i = 1, \ldots, M$. M is the number of oscillators, ω_i the intrinsic frequency of the oscillators uniformly random distributed in $[-\pi, \pi]$. K is the coupling strength of the model, A the adjacency matrix of 12 nodes shown in Fig. 3.10(a).

Having a closer look at Eq. (3.17), it becomes evident that the phases of the oscillators are influenced by the phases of other oscillators in the network. In essence, if an oscillator is moving faster, the presence of other oscillators tends to slow it down, and vice versa. Over time, this interaction can lead to synchronisation between oscillators, depending on the value of the coupling strength, K.

Figure 3.13(a) shows the MLE over coupling strength, K in $[2, 10]$. As some MLEs are less than or equal to zero and the y -axis is in logarithmic scale, there are some missing values in the plot that correspond to negative MLEs. The MLEs reach their maximum value, 0.03625, when $K = 10$, shown by a red dot in Fig. 3.13(a). Figure 3.13(b) shows the evolution of MLE over time for the trajectory where $K = 10$. To stabilise the MLE

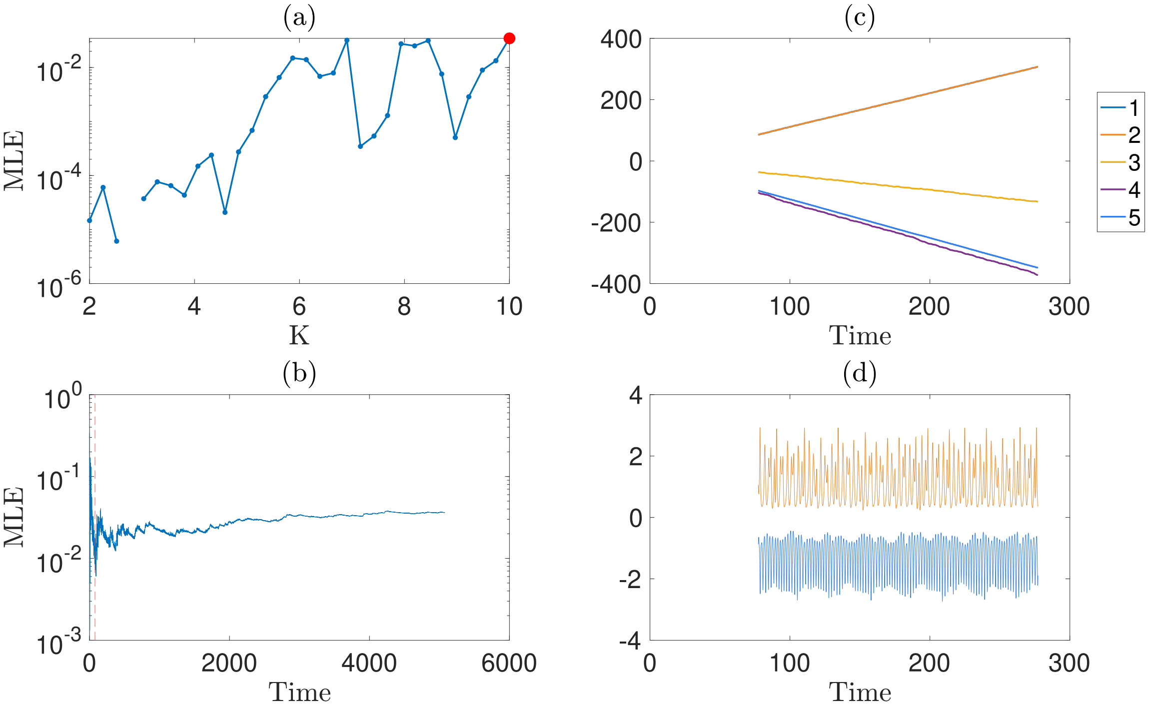

Figure 3.13: **Dynamics of coupled Kuramoto Oscillators.** (a) MLE of the deterministic Kuramoto system over the coupling strength, K. Negative MLEs are not shown in the plot because the vertical axis is in log-scale, where K in $[2, 10]$. (b) The evolution of MLE for $K = 10$, the trajectories before red-dash line is considered as transient period because it gets stabilised falling its standard deviation below predefined threshold 10^{-2}. (c) Kuramoto phases of first 5 oscillators as a function of time for $K = 10$. (d) The instantaneous frequencies of the phases of the first 2 oscillators (see detail in Sec. 4.3.3)

considering the standard deviation of 500 MLEs within a sliding time window, when it becomes less than the predefined value, 10^{-2}, the trajectory is started to be recorded. The trajectory until $t = 77.2$ is considered as a transient period following the data generation process in Sec. 3.3.2.2. Figure 3.13(c) shows the phases of the first 5 oscillators recorded after the transient period. Since the phases follow a linear trend, as shown in Fig. 3.13(c), the instantaneous frequency, defined as the time derivative of the phases, is used as a probe to investigate the nonlinear relations between pairs of oscillators (see details in Sec. 4.3.3). The instantaneous frequency has been shown in Fig. 3.13(d) for the first 2 oscillators.

3.3.3 Stochastic Dynamical Systems

Stochastic differential equation (SDE) have found widespread applications in diverse fields, from economics to biology, as they allow the consideration of unpredictable influences in the system. The general form of an SDE is as follows

$$dx_t = F(t, x_t)dt + G(t, x_t)dW_t,$$

where $F(t, x_t)$ is the drift rate function, $G(t, x_t)$ is the diffusion function, and W_t is a Wiener (or Brownian) process. Here, where $W_{t+1} - W_t \sim N(0, 1)$ and W_t are independent of the previous steps

To solve stochastic differential equations analytically, Ito's calculus can be used, which has different rules compared to classical calculus, due to the non-differentiability of these systems at any given point. Although delving into the details of stochastic calculus is beyond the scope of this thesis, numerical integration methods such as Euler-Maruyama, Milstein, or Runge-Kutta can be used to generate trajectories from SDEs. We implement the Euler-Maruyama method with its algorithm 4 given in [35].

Algorithm 4 Algorithm for Euler-Maruyama

```
Input drift function $F(t, x_t)$
Input diffusion function $G(t, x_t)$
Input initial values $t = t_0$ and $y = y_0$
Input step size $h$ and number of iterations, $n$
for <j=1:n> do
    $Y_{j+1} = Y_j + F(t_j, y_j)h + G(t_j, y_j)\Delta W_j$        ▷ where $\Delta W_j = W_{j+1} - W_j$
end for
Return $t$ and $y$.
```

3.3.3.1 Stochastic Coupled Kuramoto Oscillators

The only difference in the equation of motion for the stochastic Kuramoto model in Eq. (3.18) compared to the deterministic Kuramoto model is the inclusion of the Wiener process, as shown in the following equation

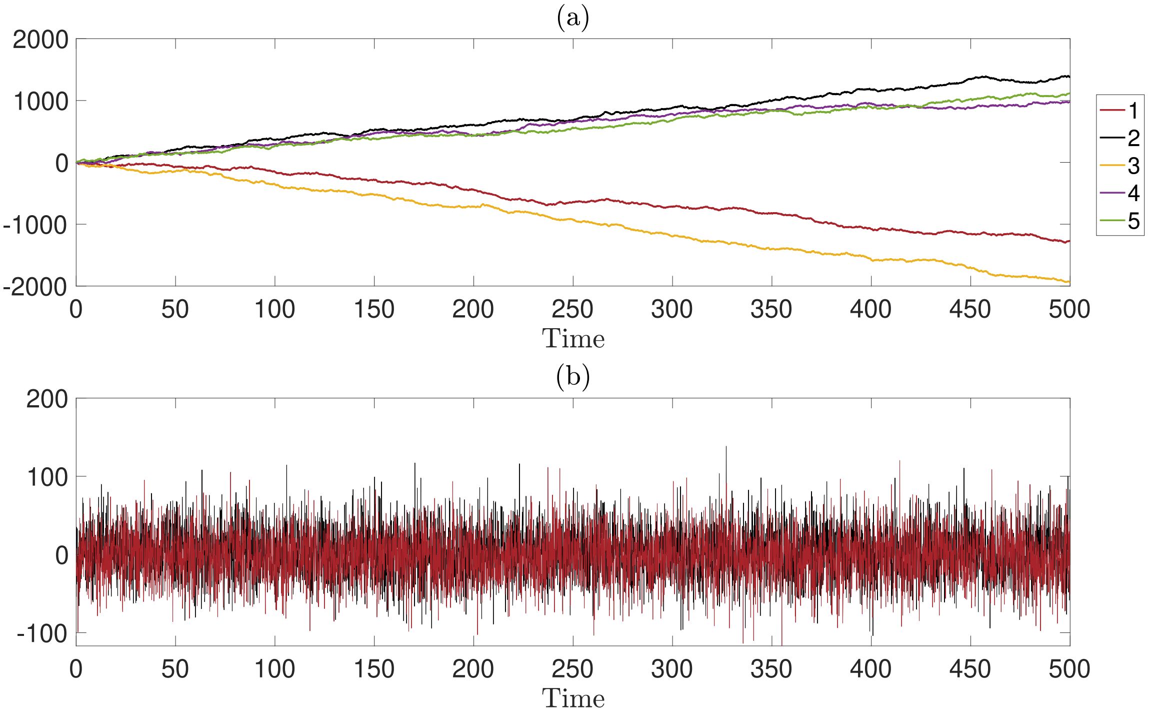

Figure 3.14: **Dynamics of coupled stochastic Kuramoto oscillators** in Eq. (3.18). (a) The evolution of phases of the first 5 oscillators over time for $D = 10$ and $K = 10$. (b) The evolution of instantaneous frequency of the first 2 stochastic oscillators.

$$d\theta_i = \omega_i dt + \frac{K}{M}\sum_{j=1}^{M} \mathbf{A}_{i,j} \sin(\theta_j - \theta_i)dt + DdW_t^i, \tag{3.18}$$

where w_i is the internal frequency of the node i, randomly chosen in $[-\pi, \pi]$ leading non-identical oscillators. D is the stochastic strength and W_t^i the Weiner process for the i^{th} node at time t, where $W_{t+1} - W_t \sim \mathcal{N}(0, 1)$ and W_t are independent of the previous steps. $\mathbf{A}$ is the adjacency matrix of 12 nodes shown in Fig. 3.10. The trajectory has been generated by Eq. (3.18) with the step size 0.1, $D = 10$ and final integration time $t_f = 500$. The initial conditions are randomly chosen in $[0, 2\pi]$, as shown in Fig. 3.14(a). The fluctuations in the trajectory are an effect of the Weiner process. However, it will not be enough to infer the network topology from the phases because of their linearity, therefore, the instantaneous frequencies will be used as a probe to infer the structure of networks, as shown in Fig. 3.14(b).

3.4 Summary

This chapter is dedicated to discussing the theoretical framework of the systems we study, focusing on their dynamics. In particular, we briefly introduce dynamical systems and chaos theory, with applications including discrete time systems such as logistic and circle maps, continuous time systems such as the Lorenz, the HR, and the deterministic Kuramoto systems, and stochastic models such as stochastic Kuramoto oscillators.

Subsequently, unless otherwise stated, we will use a modified version of RK45 with tolerance level, 10^{-8}, to solve continuous-time systems. This modification entails forcing the built-in `ode45` function in Matlab to compute solutions at an exact time step, as RK45 is an adaptive step size method. This approach enables us to compute accurate solutions of the systems and track the evolution of Lyapunov exponents at each time step. Furthermore, we computed the standard deviation of MLE of the trajectory within a sliding window of 500 time steps to ensure that we exclude the transient period from the solution. In the plot of the trajectories over time, the vertical red dashed line indicates the time when the standard deviation of MLE calculated over 500 points falls below the predefined level. Therefore, we consider the trajectory before this line as the transient period.

Since the dynamics of the systems depend on various variables such as network structure, model parameters, and coupling strength, we will present the maximum Lyapunov exponents (MLEs) of the systems studied in the following chapters as evidence of chaos.

Chapter 4

Chaotic Synchronisation

4.1 The Concept of Synchronisation

Synchronisation is a natural phenomenon that has been attracting the interest of researchers for a long time. Early discussions about synchronisation can be found in the scientific literature, such as the case of the synchronisation of wall clocks investigated by C. Huygens in the 17*th* century. In a letter he sent to his father, he shared his observation that the wall clocks hanging on the same platform were moving together after a while. This was an example of PS of weakly coupled oscillators. Later, he continued his work and invented more precise clocks for the board of ships, which was a very challenging problem at that time. He also created a clock for a church, guaranteeing a deviation of at most 8 minutes per week [51].

Another case of synchronisation in the early 1900s was observed in fireflies in Southeast Asia, as recorded by American travellers in their diaries. The fireflies were seen flashing together. Initially, this phenomenon was thought to be an illusion and was not investigated scientifically. With the development of cameras, the reality of synchronised firefly flashing became undeniable. This phenomenon intrigued researchers, prompting them to explore the synchronisation of fireflies. There are many questions to ask: “What is the mechanism behind this peculiar behaviour? How did the fireflies achieve synchronisation? Was there a “maestro” or alpha individual that others followed?” The reasons for this behaviour and the underlying biological mechanism remain unclear; however, mathematical models that simulate firefly synchronisation have been developed over time, with contributions from mathematicians, such as Wiener and Winfree [61]. In 1975, Y. Kuramoto published his seminal paper [37], which introduces the so-called Kuramoto model given by Eq. (3.17). As an interpretation, we can say that there is no “maestro” among the group of fireflies, which drives all in harmony. Instead, all fireflies have an effect on the ones nearby; if some run fast, they reduce their speed; otherwise, they speed up. Interestingly, the fireflies’ flashes are synchronised after some time.

While the notion of units in chaotic dynamics reaching synchrony may appear controversial, such synchronisation can be observed under various configurations. Complete synchronisation represents the simplest form of synchronisation observed in chaotic trajectories in a drive-response system. In this context, one of the variables in the response system is driven by the variables of the chaotic drive system. If all Lyapunov spectra are negative in the response system, the observation of CS becomes possible. Additionally, CS manifests in bidirectionally coupled systems, where the dynamics is chaotic. The authors in [15] investigate various types of synchronisation in chaotic trajectories, including PS, IPS, lag synchronisation (LS), intermittent lag synchronisation (ILS), generalised synchronisation (GS), and almost synchronisation (AS). In PS, the phases of signals may synchronise even when their amplitudes are weakly correlated. In IPS, signal phases slip within a synchronous regime. Lag synchronisation involves the locking of amplitudes and phases in the presence of a time lag τ_{lag}. If LS occurs most of the time, but intermittent burst behaviour exists, ILS is observed. Generalised synchronisation can occur between entirely different systems if their trajectories can be associated with a function. Finally, AS occurs if the difference between the trajectories of two systems lies within an asymptotically bounded region.

This chapter starts by discussing CS in both directional and bidirectional coupling scenarios. An example of CS between two nodes in a 5-node network is presented in Sec. 4.2.3 using data generated from a coupled logistic map. Before delving into phase synchronisation, we explore the concepts of amplitude, phase, and frequency in a signal, accompanied by a brief discussion of the Fourier transform, the spectrogram, and the Hilbert transform. The analytic signal approach, introduced by Gabor in 1946 [23], is employed to derive instantaneous amplitudes and phases from chaotic signals. Subsequently, an example of PS and IPS is observed in the trajectories of 5 coupled Rössler systems. To quantify amplitude or phase synchronisation in the trajectories, we used the amplitude order, the Kuramoto phase order, and pairwise phase order and implemented them in the cases of CS, PS and IPS.

4.2 Complete Synchronisation

In the realm of CS, it signifies that the trajectories of two systems move together in synchrony under a specific coupling configuration, even as the systems exhibit chaotic behaviour and remain sensitive to initial conditions. In the Pecora & Carroll configuration, a subsystem (referred to as the slave) is driven by a variable from a master system, where the entire system displays chaotic behaviour. However, for synchronisation to occur, all conditional Lyapunov exponents of the slave system must be negative. In the case of bidirectional coupling configuration, two chaotic trajectories might be synchronised under the reciprocal influence of each other.

The mathematical definition of CS is given as follows.

Definition 4.2.1 (Complete Synchronisation)**.** *For two time series signals, x_t and y_t, they are called "completely synchronised" if the following condition meets.*

$$\lim_{t\to\infty} |x_t - y_t| = 0.$$

4.2.1 Drive-response Systems

Following [15], the first configuration studied to occur CS is called the drive response (or master-slave system in Pecora & Carroll configuration, as one of the variables of the master system drives the variables of the subsystem. We used three variables of the Lorenz system in Eq. (3.15) as master and two of them, except the driver, as a slave system. When we generate the replicas of subsystem with different initial conditions, CS occurred between the variables of subsystem and its replicas if the conditional Lyapunov exponents of the subsystem is all negative. When the x variable is chosen as a driver, the equation of slave system,

$$\begin{aligned} \dot{y}_1 &= (\rho - z_1)x - y_1 \\ \dot{z}_1 &= xy_1 - \beta z_1, \end{aligned} \tag{4.1}$$

where $\sigma = 10$, $\rho = 28$ and $\beta = \frac{8}{3}$. In this slave system, we generate data for the variables y_1 and y_2, as well as their replicas y_1' and z_1' using different initial conditions. Figures 4.2(a) and (b) show that variables y_1 and z_1 quickly synchronise with their replicas y_1' and z_1', respectively.

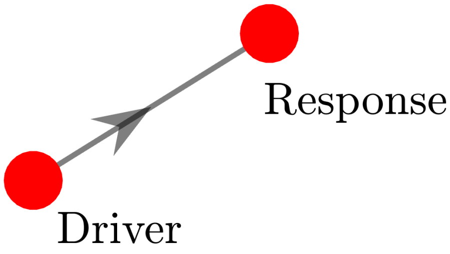

Figure 4.1: **Scheme of drive-response System.** Plot visualises the directional interaction between driver and response systems in (4.1), (4.3) and (4.2)

However, if we choose the z variable as a driver, the equation of slave system,

$$\begin{aligned} \dot{x}_1 &= \sigma(y_1 - x_1) \\ \dot{y}_1 &= (\rho - z)x_1 - y_1, \end{aligned} \tag{4.2}$$

and Fig. 4.2(c) demonstrates that the absolute difference between x_1 and x_1' as well as y_1 and y_1' increases so, slave system and its replicas cannot achieve CS. The reason of that is MLE of the slave system, when it is driven by z variable, is positive as shown in Table 4.1. By the conditional Lyapunov exponents, we mean that only the evolution of Lyapunov vectors in the slave system is considered; in other words, we measure the exponential divergence (or convergence) of the subsystem only. When the subsystem is driven by y variable, response variables x_1 and z_1,

$$\begin{aligned} \dot{x}_1 &= \sigma(y - x_1) \\ \dot{z}_1 &= x_1 y - \beta z_1, \end{aligned} \tag{4.3}$$

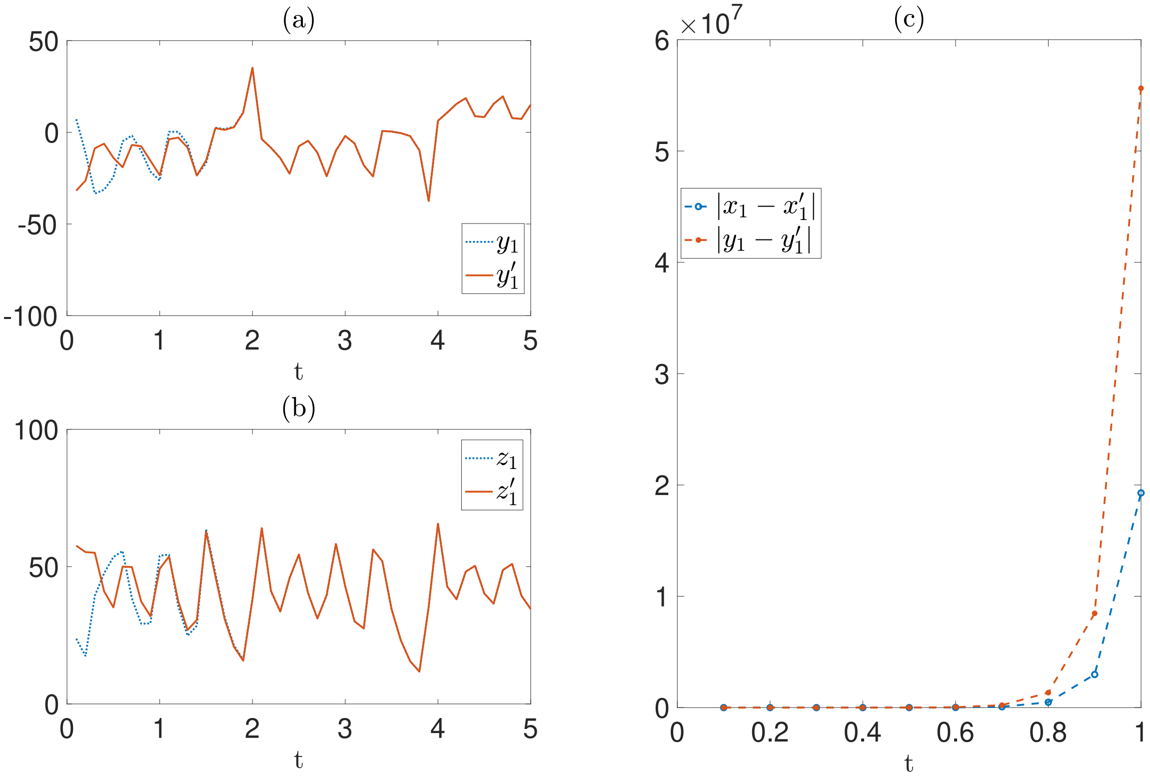

Figure 4.2: **CS in drive-response system.** When the slave system is driven by x variable, (a) shows that CS occurred between y_1 and its replica, y_1', for different initial condition, (b) CS occurred between z_1 and its replica for different initial condition, z_1'. (c) When the system is driven by z, the absolute difference between x_1 and its replica x_1', as well as between y_1 and y_1', increases, indicating that the subsystem does not achieve CS.

which synchronises with their replicas since there is no positive conditional Lyapunov exponent.

System	Drive	Response	Conditional Lyapunov Exponents
Lorenz	x	(y, z)	$(-2.43, -2.57)$
$\sigma = 10$, $\rho = 28$, $\beta = \frac{8}{3}$	y	(x, z)	$(-4.00, -16.00)$
	z	(x, y)	$(0.0001, -11.00)$

Table 4.1: The conditional Lyapunov Exponents of the subsystems (4.1), (4.3), and (4.2), respectively from top to bottom.

4.2.2 Bidirectionally Coupled 2-Dimensional System

In addition to the drive-response system, CS can also occur in bidirectionally coupled systems. Following the discussion in [15], we use all variables coupled Lorenz system given by

$$\begin{aligned}\dot{x}_i &= \sigma(y_i - x_i) + K\sum_{j=1}^{2}(x_j - x_i),\\ \dot{y}_i &= (\rho - z_i)x_i - y_i + K\sum_{j=1}^{2}(y_j - y_i),\\ \dot{z}_i &= x_i y_i - \beta z_i + K\sum_{j=1}^{2}(z_j - z_i),\end{aligned}$$

where $\sigma = 10$, $\rho = 28$, $\beta = 8/3$ and coupling strength $K = 2$, for $i = 1, 2$. As seen in Fig. 4.3(a), (c), and (d), all variables of the model are in synchrony after a while, however, the whole system shows chaotic behaviour with positive MLE, which is about 0.9052. The three-dimensional phase space of the variables x_1, y_1, z_1 belonging to the 1^{st} oscillator forms the usual butterfly shape as shown in Fig. 4.3(b).

In this configuration, the strong enough coupling leads the CS of the variables, which means that the coupling strength, K, should be higher than the MLE of uncoupled model. Figure 4.3(e) shows that the Euclidean distance between the variables,

$$e = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2},$$

as a function of rate of K with MLE. The mean of the Euclidean distance between the points (x_1, y_1, z_1) and (x_2, y_2, z_2) over time becomes zero when the condition $K \geq$ MLE holds.

4.2.3 Multidimensional Systems

In our further investigation, we observe that the CS might occur between some pairs of a system even if it has more than 2 oscillators, which is the case that we will study in the network inference later. Therefore, we generate the trajectories from the coupled

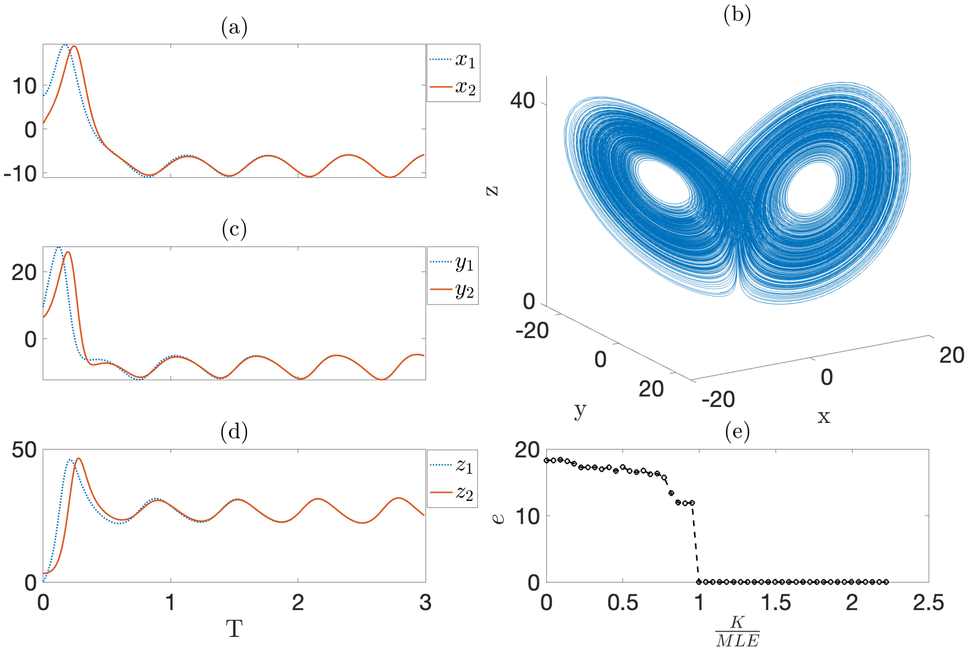

Figure 4.3: **CS in all coupled Lorenz system.** For $K = 2$, CS occur between the variables (a) x_1 and x_2, (c) y_1 and y_2 (d) z_1 and z_2. (b) plot of the phase space of the 1^{st} oscillator in three-dimensional space for $K = 2$, indicating chaotic dynamics with usual butterfly-like shape and MLE = 0.9026. (e) The function of mean Euclidean distance between points (x_1, y_1, z_1) and (x_2, y_2, z_2). The CS occurs if the condition $K \geq$ MLE holds where MLE is for the uncoupled system, circa 0.9026.

logistic map using the Eqs. (3.6) and (3.13), with a coupling strength, 0.2, based on the interaction in the network of 5 nodes in Fig. 4.4(a). Figure 4.4(b) demonstrates that the evolution of MLE, which converges to a positive value as iterations approach infinity, is evidence of chaos. After 500 MLEs within the sliding time window is stabilised, becoming less than the predefined value 10^{-3}, the time series starts to be recorded at the time indicated by the vertical red dashed line. The time length of the trajectory after the transient period is 10^5. Figure 4.4(c) shows the cumulative absolute difference between the trajectories of some pairs. Around 6×10^4 iteration, the cumulative difference between pairs of 1 and 3 becomes constant. In Fig. 4.4(d), we demonstrate the log-scaled absolute difference between the trajectories of these nodes, indicating that the nodes achieve CS around iteration 6×10^4. In the sense of network inference, due to this synchronisation, their similarity measures become high, as the two nodes behave as one (see discussion in

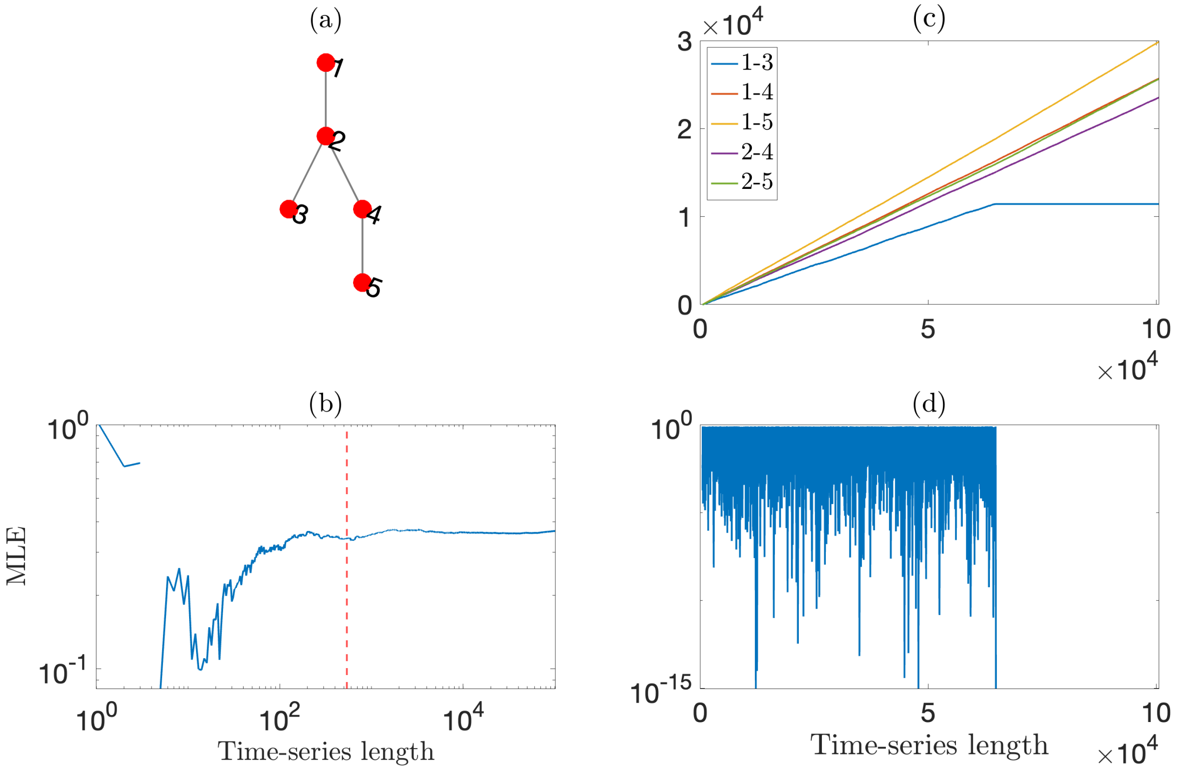

Figure 4.4: **CS in multidimensional coupled logistic map.** (a) shows the network of 5 nodes used in the coupled logistic map, (b) shows the evolution of MLE over time in both log-scaled axes, indicating chaotic dynamics, (c) shows the cumulative difference between some pairs, CS occurred between node 1 and 3 around 6×10^4 iteration. (d) The absolute difference between nodes 1 and 3. The zero error after 6×10^4 iterations is not plotted as a result of the log scale in y-axis.

Sec. 5.4.5.1). This synchronisation hinders the inference of network structure from the data, necessitating the detection of such behaviour and awareness of its impact on network inference.

4.3 Phase Synchronisation

Before delving into the details of phase synchronisation, we first define what phase means in various contexts. In general, from the perspective of signal processing, amplitude, phase, and frequency are three crucial concepts for characterising a signal. While they gain a physical interpretation in circular motion (or in a sine wave) in contrast to chaotic trajectories, we will initially focus on their physical interpretations in such systems. Generalising these concepts to chaotic signals is useful for signal characterisation, but they are ill defined [13, 14, 31]. We discuss the amplitudes, phase, and frequency of chaotic signals using the analytic signal approach introduced by Gabor in 1946. Finally, we define PS and present an example of PS and IPS using the coupled Rössler system from a network of 5 nodes.

4.3.1 Circular Motion & Sine Wave

The position of an object following a circular path around a centre refers to its *phase*, denoted by θ. For example, in Fig. 4.5(c), the red circle represents an object following the blue circular path centred at the origin with radius 2. In this demonstration, the phase of the object is $\frac{\pi}{4}$ as it is the angle between the line connecting the object to the origin and the x-axis. The rate of change of θ with respect to time, t, gives its angular velocity, $\omega = \frac{d\theta}{dt}$. The projection of the position on y-axis gives the sine wave as a function of time in Fig. 4.5(a) and its equation is given by,

$$x(t) = A\sin(2\pi f t), \tag{4.4}$$

where amplitude, $A = 2$, frequency, $f = 2$ and $t \in [0, 2]$. f indicates the number of cycles per second; therefore, the wave oscillates 4 times in 2 seconds. The period $T = \frac{1}{f}$, refers to the time that passes for one cycle, which equals $\frac{1}{2}$. As phase, $\theta = 2\pi f t$ and angular velocity, $\omega = \frac{d\theta}{dt}$ at any time t, we can deduce the relation between frequency and phases as follows

$$\omega = \frac{d\theta}{dt} = 2\pi f \implies f = \frac{1}{2\pi}\frac{d\theta}{dt}. \tag{4.5}$$

In Fig. 4.5(b), blue lines demonstrates the unwrapped phases of the signal in $[0, 2\pi]$, as a function of time, using the fact that $\theta = 2\pi f t$. It is more common to use wrapped phases to obtain a continuous phase function of time, but here, we prefer to show it in $[0, 2\pi]$. On the other hand, orange line shows the phases deduced from the concept of analytic signal using HT (see Sec. 4.3.3), we get $\frac{\pi}{2}$ radian shifted phases as a result of HT [46].

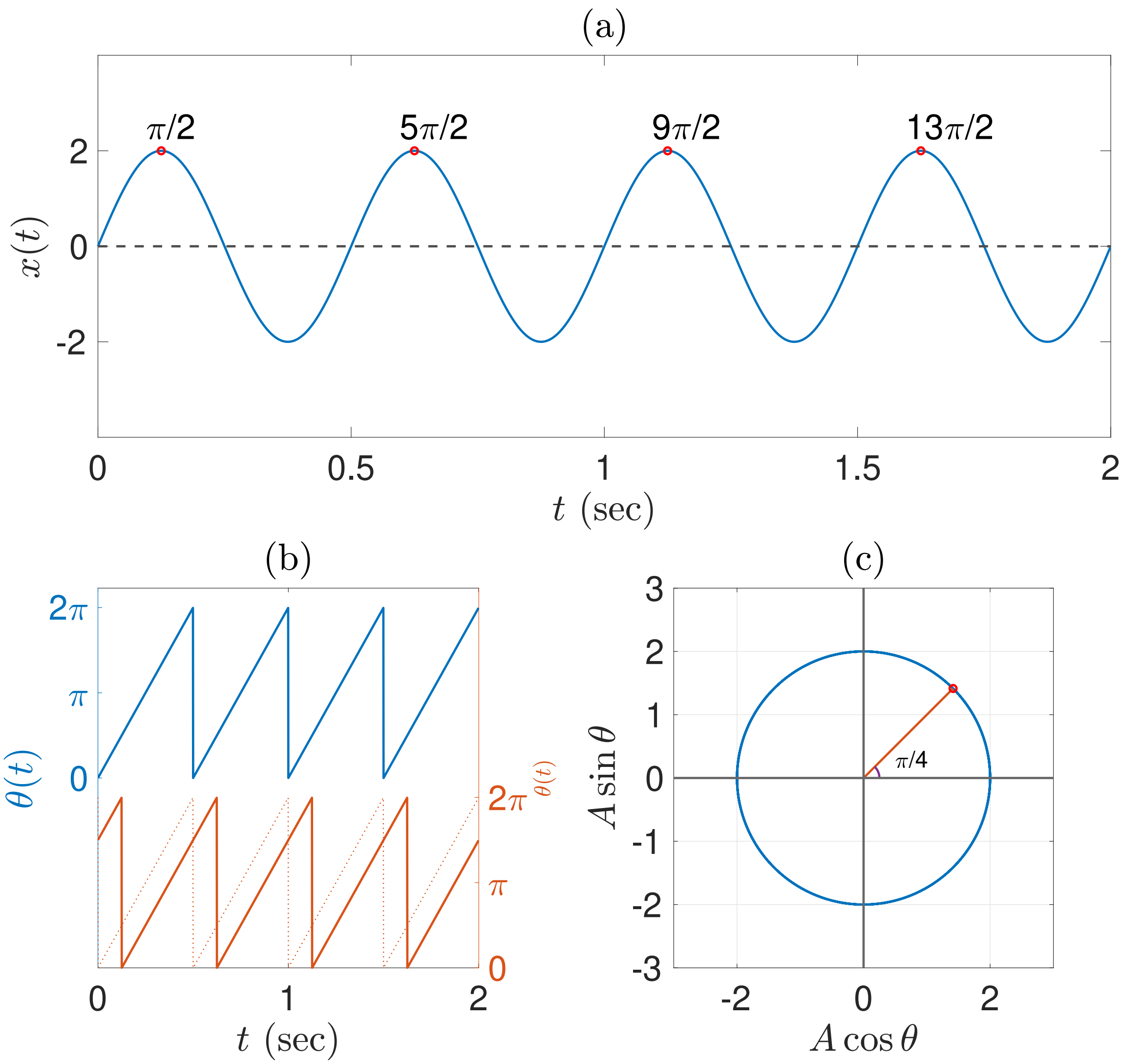

Figure 4.5: **Phase in circular motion** (a) Sine wave as a function of time (b) Phase of the signal as a function of time, where blue indicates the angle in the sine function and orange line is the instantaneous phases computed from HT, obtaining $\frac{\pi}{2}$ radian phase shift (c) Circular motion of an object shown by red circle at phase $\frac{\pi}{4}$.

4.3.2 Fourier Transform & Spectrogram

To characterise the signal in the sense of its phase and frequency, it is essential to discuss Fourier transform before delving into the concept of an analytic signal introduced by Gabor [23], as they collectively construct a basis. Fourier transform of continuous-time signal $x(t)$, [45, 46],

$$X(f) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{\infty} x(t) e^{-2\pi i f t} dt,$$

where $\mathcal{F}\{\cdot\}$ is the Fourier operator. By replacing "integral" with "sum", we can obtain Fourier transform (FT) for discrete-time signal,

$$X(f) = \sum_{t=-\infty}^{\infty} x(t)e^{-2\pi ift}, \tag{4.6}$$

where $X(f)$ is a complex-valued function of frequency, which we can express in rectangular form,

$$X(f) = X_R(f) + iX_I(f) \tag{4.7}$$

where X_R denotes the real part, X_I the complex part of $X(f)$. We also express it in polar form,

$$X(f) = |A(f)|e^{i\theta(f)} \tag{4.8}$$

where $A(f)$ is called Fourier amplitude in the frequency domain, which is equal to the modulus of the complex function, $|X(f)| = \sqrt{X_R(f)^2 + X_I(f)^2}$ and, $\theta(f)$ the Fourier phase,

$$\theta(f) = \tan^{-1}\frac{X_I(f)}{X_R(f)}.$$

The FT is commonly used to identify the dominant frequencies of a signal, reduce any present noise, and/or filter the frequency to focus on a specific interval. It is also capable of decomposing a composite signal. For instance, consider a signal composed of three sinusoidal waves,

$$x(t) = 0.5\sin(2\pi f_1 t) + 0.7\sin(2\pi f_2 t) + \sin(2\pi f_3 t),$$

where $A_1 = 0.5$, $A_2 = 0.7$ and $A_3 = 1$, are the amplitudes and $f_1 = 10$, $f_2 = 20$, $f_3 = 30$ the frequencies of the sine waves. We introduce Gaussian noise, γ, to the signal, $x(t)$, to obtain the noisy signal, x_s.

$$x_s(t) = x(t) + \gamma,$$

where $\gamma \sim \mathcal{N}(0, 4)$. Figure 4.6(a) and (c) shows the signal, $x(t)$ and the noisy signal, $x_s(t)$, in the time domain, respectively and Fig. 4.6(b) and (d) demonstrate the FT of the corre-

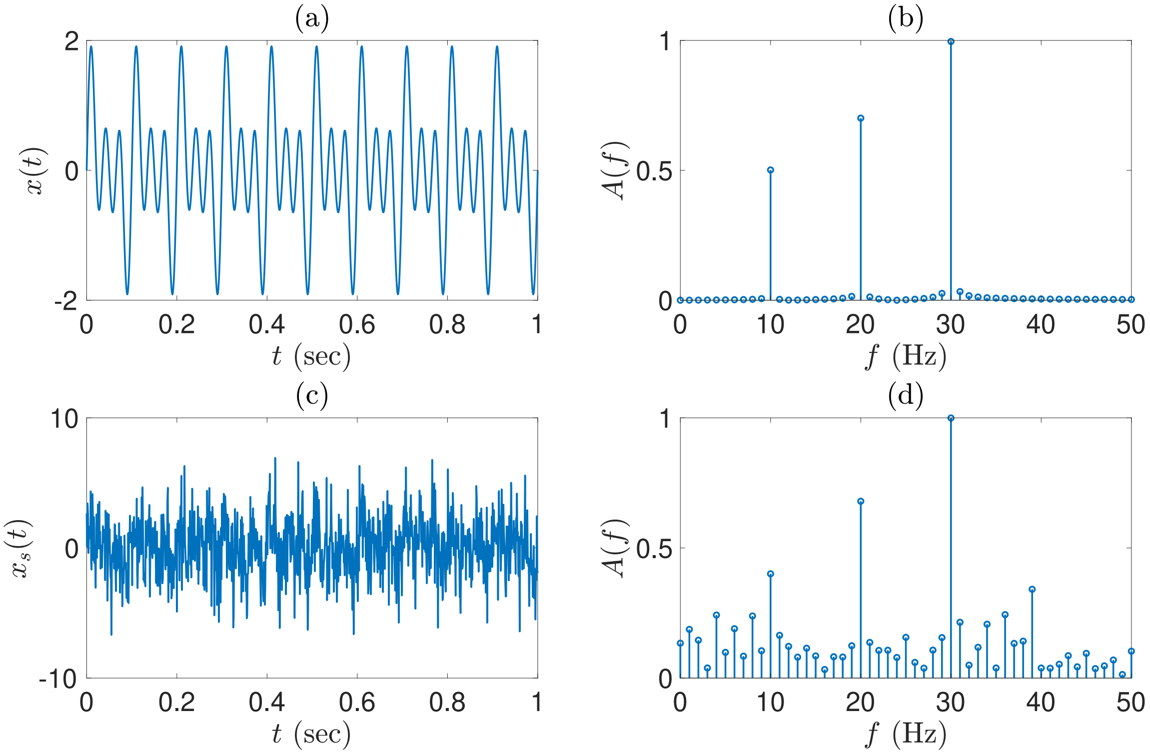

Figure 4.6: **The FT of composite and noisy signal** (a) The composite signal of three sinusoidal waves, $x(t)$, in the time domain. (b) The Fourier amplitude, $A(f)$ of the signal $x(t)$ as a function of frequency, f. (c) The noisy signal, $x_s(t)$, in the time domain (d) The Fourier amplitude of the noisy signal $x_s(t)$, in the frequency domain. This shows the FT is able to capture dominant frequencies even if the noise is present.

sponding signal in the frequency domain, where the x-axis shows dominant frequencies, f_i and y-axis the amplitude of the signals, A_i for $i = 1, 2, 3$, where the amplitudes are equal to the mean of the modulus of the complex-valued signal, $X(f)$. Despite the significant noise is present in the data, its dominant frequencies are the same as those of the original signal. This allows us to eliminate noise from the data by filtering out the small frequencies and restore the original signal through inverse FT.

Although FT is a useful tool for the frequency analysis of a signal, composed of signals with constant frequencies, if the frequency is time varying, the FT is not able to show changes over time. In such a case, FT of the signal in sliding time window can be investigated by the spectrogram. A signal with frequency varying over time, x_t,

$$x_t = 0.5\sin(2\pi f_1 t),$$

where $f_1(t) = 0.5t$ meaning that the frequency of the signal is the half of the current time, t, and it increases over time as shown in Fig. 4.7(a). The Fourier transform of the signal cannot capture the variation frequency in the frequency domain (see Fig. 4.7(b)) because none of the frequencies is superior to others. However, the spectrogram considering the frequency in the time-sliding window gives a better understanding of the frequency-varying signal. Figure 4.7(c) demonstrates that the dominant frequency increases linearly over time, where the power spectra of the signal, $|X(f)|^2$, which determines how the energy is distributed in the frequency domain, indicated by the colour map, showing that the dominant frequency lies along the yellow band. The instantaneous frequency computation from HT using Eq. (4.5), shows similar pattern with spectrogram.

In conclusion, a composite signal with constant frequencies can be decomposed by FT, proves invaluable in decomposing a composite signal with constant frequencies. This enables the identification of dominant frequencies and facilitates the reduction of noise present in the system. Such signals are termed “stationary”, as they exhibit consistent statistical properties, such as mean and variance, over all time intervals. Conversely, frequency-varying signals are nonstationary, which results in the absence of a dominant frequency. Next, we will explore frequencies for both regular and chaotic signals.

4.3.2.1 Frequency Analysis of Regular and Chaotic Signals

In this section, we apply FT to analyse both 4-periodic and chaotic dynamics iterated from the logistic map with $r = 3.5$ and $r = 4$, respectively, as seen in Fig. 4.8. We assumed a sample rate of $F_s = 100$ Hz (corresponding to a time step $T = \frac{1}{F_s} = 0.01$ seconds) and a time series length of 10^4. For the 4-periodic motion, with a frequency of 25 Hz, meaning that the signal repeats the same pattern 25 times in a second, as shown in Fig. 4.8(a) and (b). However, unlike the periodic signal, the chaotic signal does not exhibit a single

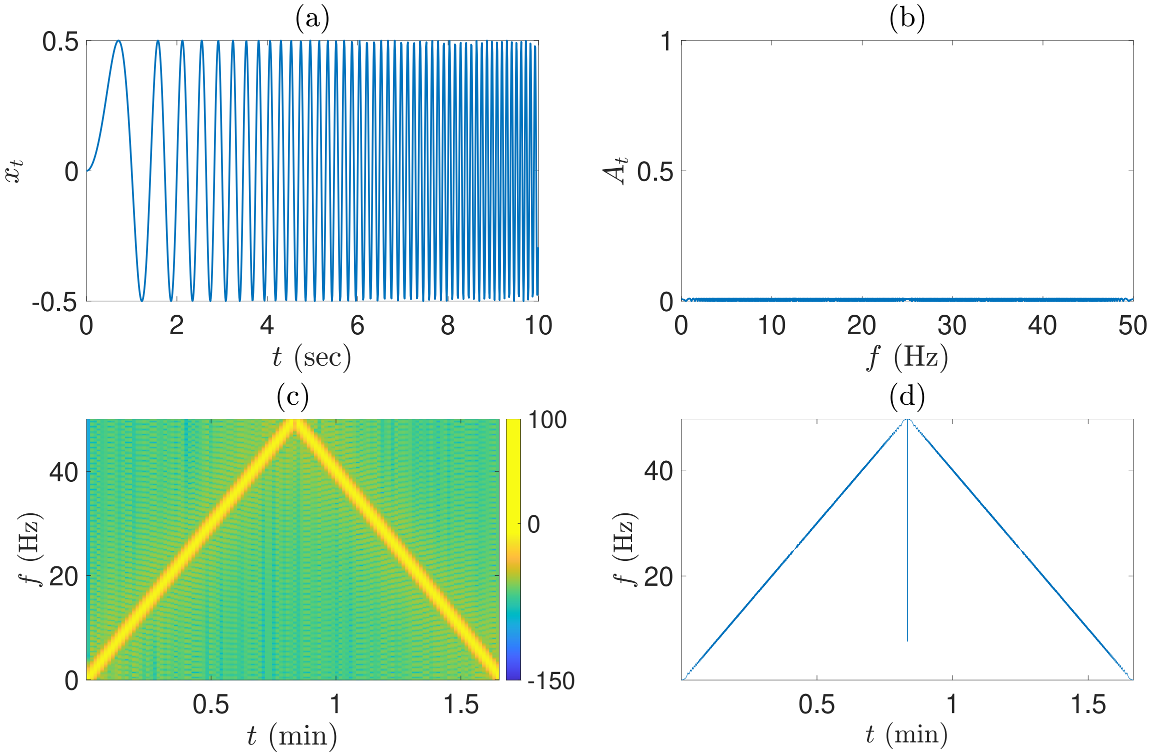

Figure 4.7: **Frequency-varying signal**. (a) The signal as a function of time. (b) FT of the signal in the frequency domain, where no dominant frequency appears. (c) Spectrogram of the signal considering the FT in sliding time windows of 100 data points. Power spectra of the signal in decibels, $|X(f)|^2$, indicated by the colour map. (d) Instantaneous frequency calculated from the concept of analytic signal using the HT and Eq. (4.5).

dominant frequency, as shown in Fig. 4.8(c) and (d).

4.3.3 Analytic Signal

Once one tries to generalise the definitions of amplitude, phase, and frequency in circular motion to nonstationary signals, following the discussions in [13, 14, 30, 31], the physical interpretation of the concept of instantaneous frequency for chaotic signals becomes blurred. Let us consider the concept of frequency. In a periodic motion, it refers to the number of cycles or repetitions in a unit of time. Even if the FT allows us to decompose the signal to its sine components, nonstationary signals cannot be decomposed well [13]. Even though there is no clear physical interpretation of the frequency of a nonstationary signal, and it seems controversial to think in the way of periodic motion, it is still useful to characterise

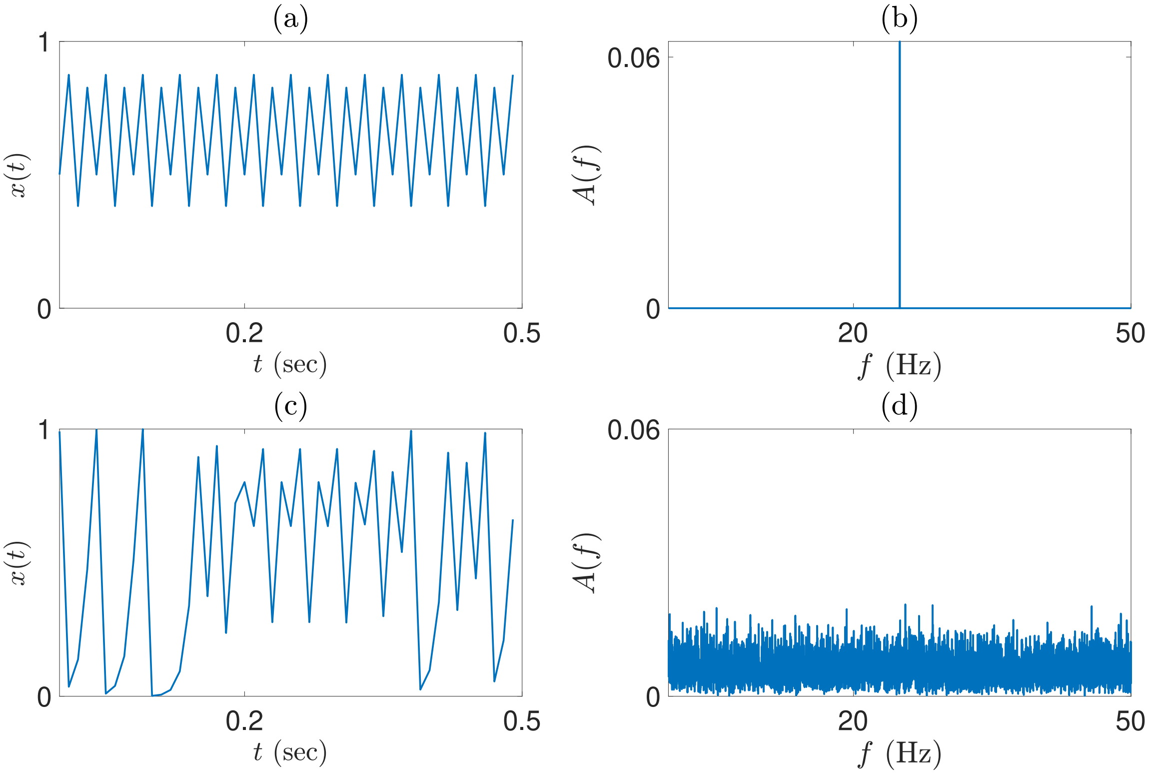

Figure 4.8: **The FT of regular and chaotic signals** (a) 4-periodic signal generated from logistic map with $r = 3.5$ and (b) its frequencies. (c) Chaotic signal generated from the logistic map with $r = 4$ and (d) its frequencies.

the signal. We use the term “instantaneous” to identify these characteristics of the signal for nonstationary signals.

To measure instantaneous phases, the authors in [51] employ the concept of Poincaré Map for systems with nearly circular cross sections within subdimensional space, but this approach cannot be generalised to all types of signal. The authors in [30] discussed that HT is superior to FT in chaotic signals in terms of its uniqueness, but introduced an intrinsic mode function to overcome the issue of negative frequency. Unfortunately, there might be various ranges of intrinsic mode functions defined. Consequently, we are actually interested in the information transferred among the nodes in a network through their amplitudes and phases (see Chapter 5), we will compute them using the concept of analytic signal introduced by Gabor in 1946 [23] and instantaneous frequency from the instantaneous phases using Eq. (4.5) as it allows us to reveal connectivity from time-series data.

To obtain the instantaneous amplitudes and phases from a signal, Gabor introduced the concept of analytic signal based on the HT in [23],

$$\begin{aligned} z(t) &= x(t) + iH[x(t)] \\ &= A(t)e^{i\theta(t)} = \underbrace{A(t)\cos(\theta(t))}_{z_R(t)} + i\,\underbrace{A(t)\sin(\theta(t))}_{z_I(t)}, \end{aligned} \tag{4.9}$$

where $x(t)$ is the original signal, $A(t)$ the envelope (or instantaneous amplitude) of analytic signal, i complex unit, and, $H[x(t)]$ is the Hilbert transform of the signal,

$$H[x(t)] = -\frac{1}{\pi}\text{p.v.}\int_{-\infty}^{\infty} \frac{x(\tau - t)}{\tau} d\tau. \tag{4.10}$$

where p.v. denotes the Cauchy principle value of integral. The real part of the analytic signal, $z_R(t)$ equals to signal itself, $x(t)$ and one can compute instantaneous phases,

$$\theta(t) = \tan^{-1}\frac{z_I(t)}{z_R(t)}. \tag{4.11}$$

The envelope, $A(t)$, is the distance of the analytic signal to the origin in the polar coordinate and wraps the boundaries of absolute value of the signal. In the Fig. 4.9, the analytic signal and its components have been presented for the signal $x(t)$ given by

$$x(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t),$$

where $A_1 = 0.5$ and $A_2 = 0.5$ are the amplitudes of the sine waves, $f_1 = 10$ and $f_2 = 20$ are the frequencies. Figure 4.9(a) shows the signal with blue line in the time interval $[0, 1]$ seconds, which wrapped by its envelope $A(t)$ and $-A(t)$. Figure 4.9(c) demonstrates the analytic signal in the complex domain at the time that instantaneous phase θ equals to $\frac{\pi}{6}$, shown by the orange line segment. The orange line traces the blue closed curve over time; the length of it shows the instantaneous amplitude, and the angle with the x-axis gives the instantaneous phases at any time t. Figure 4.9(b) shows its function as a time. From the instantaneous phases, we compute the instantaneous frequency using the Eq. (4.5), although other approaches presented in [14].

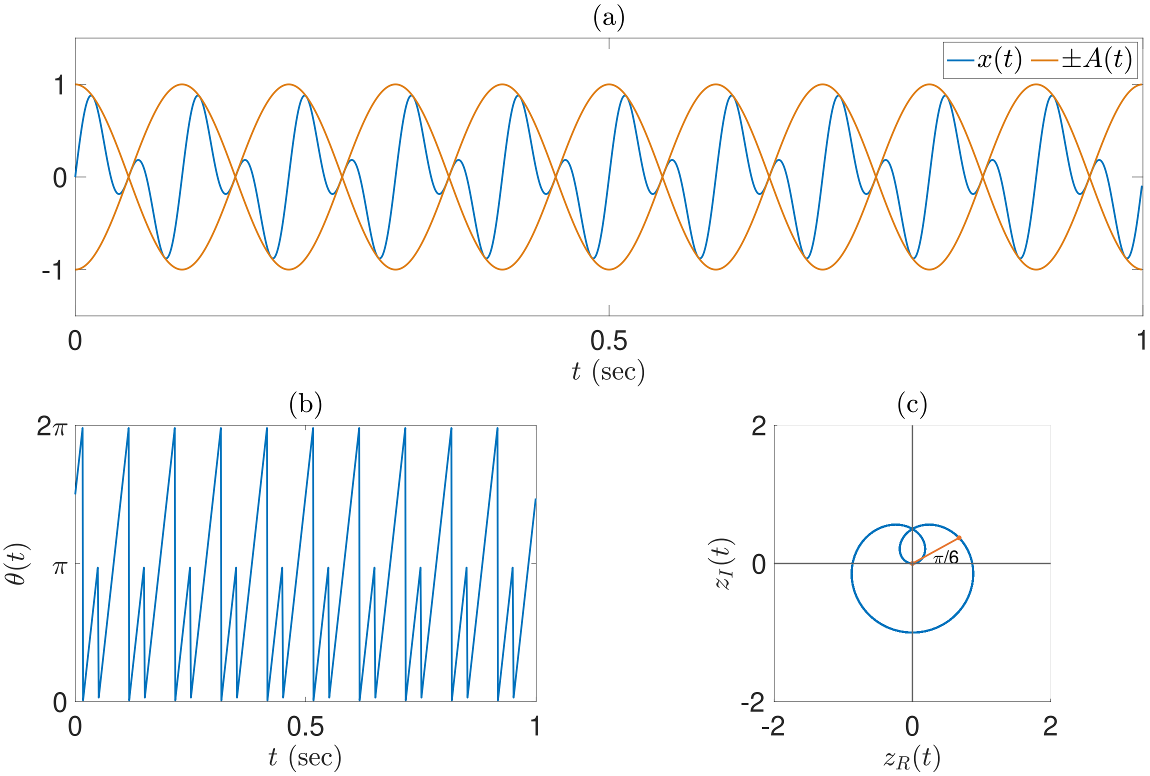

Figure 4.9: **The concept of analytic signal**. (a) The blue line shows the signal over time, wrapped by its envelope $\pm A(t)$. (b) Instantaneous phase of the signal in the same time interval, obtaining from the analytic signal. (c) the Analytic signal, $z(t)$, is presented in the polar coordinate, the projection of the orange line segment on x and y axes gives the real and imaginary part of $z(t)$, respectively.

4.3.4 An example of PS and IPS in chaotic system

As we defined three characteristics of a signal (amplitude, phase, and frequency), we can now go back to our discussion on phase synchronisation.

Definition 4.3.1 (Phase Synchronisation)**.** *Let x and y be signals with the phases ϕ_1 and ϕ_2. x and y are called their phase-synchronised signals if the linear combination of the phases stays bounded, $\psi_{m,n} = m\phi_1 - n\phi_2 < c$, where c is a constant real number, m and n positive integers.*

Due to simplicity, we consider $m : n = 1 : 1$ as the authors did in [15, 55] to show that two nonidentical chaotic oscillators of Rössler system with relatively small coupling strength, 0.035, achieve PS. We extend this system to five chaotic oscillators of Rössler system based on connectivity in Fig. 4.4(b). The coupled Rössler system is given by the set of differential equations,

$$\begin{aligned}
\dot{x}_i &= -\omega_i y_i - z_i + K\sum_{j=1}^{M}\mathbf{A}_{i,j}(x_j - x_i) \\
\dot{y}_i &= \omega_i x_i + a y_i \\
\dot{z}_i &= f + z_i(x_i - c),
\end{aligned} \tag{4.12}$$

where w_i is the intrinsic frequency randomly chosen in $[1-0.015, 1+0.015]$, which results in the non-identical oscillators. We set the parameters as $a = 0.15$, $f = 0.2$, $c = 10$ and coupling strength $K = 0.02$, and solved the system with the final integration time 10^4, and step size 0.1. We start recording the orbit after the MLE stabilises. To do so, we compute the MLE as a function of time and compute the standard deviation of a sliding time window of 500 MLEs. When the standard deviation of 500 MLEs in a window is smaller than a predefined threshold, we consider that the MLE has converged to a value. If this value is positive, it is an indication that the dynamics is chaotic, and we start recording the time series after the last of these 500 time points. In this context, the trajectory is recorded once the MLE stabilises, under the assumption that the standard deviation of 500 MLEs within sliding time windows is less than or equal to the predefined threshold, 10^{-3} so the first 3730 points are considered as transients. The system exhibits chaotic behaviour, as evidenced by a positive MLE value, 0.0765.

Fig. 4.10(a) demonstrates the pairs of oscillators achieving PS because the difference between them stay bounded for $m = n = 1$ as stated in the definition 4.3.1. We also observed IPS between the pairs 4 and 5 as their phases slip in synchronous regime. However, phases of the rest of the pairs do not achieve synchrony as shown in the Fig. 4.10(b) since the absolute difference between pairs increases over time.

In this example, we explore the phase synchronisation of $1:1$, but the values of $m:n$ might be any pair of integers. Therefore, we will introduce metrics to quantify the amplitude and/or phase order of a signal next.

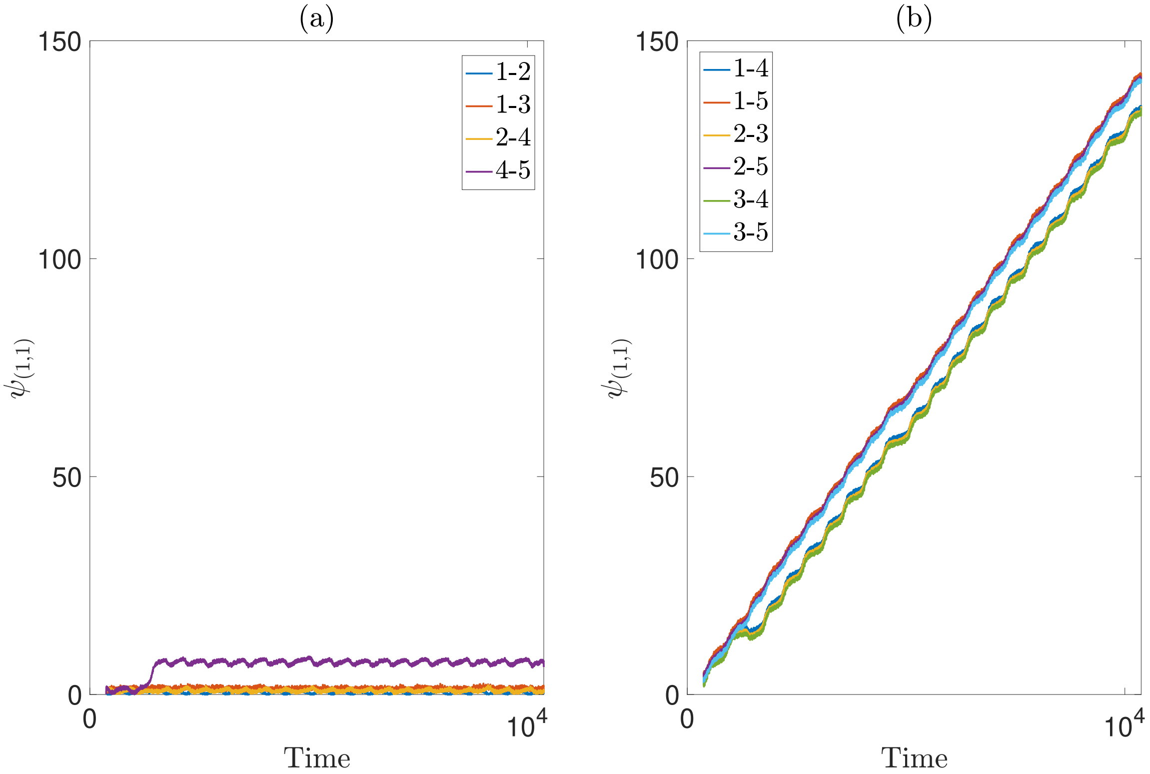

Figure 4.10: **Phase synchronisation in network of 5 nodes**. The figure shows (a) the pairs that are in 1:1 phase-locking, (b) the pairs that are not in 1:1 phase-locking.

4.4 Synchronisation Measures

4.4.1 Amplitude Order

CS between pairs of nodes can be observed by plotting the difference between them over time, but this becomes impractical for large networks due to the number of pairs. However, [57] proposes a measure that quantifies the order of magnitudes of the whole system of discrete map. The order parameter R is defined as follows.

$$R = \frac{[\langle X^2 \rangle - \langle X \rangle^2]}{\overline{[\langle {x^i}^2 \rangle - \langle x^i \rangle^2]}}, \tag{4.13}$$

where $X_n = \frac{1}{N}\sum_{i=1}^{N} x_n^{(i)}$ is the network average of n^{th} iteration, $\langle \cdot \rangle$ denotes the time average and $\overline{[\cdot]}$ the average over network. In the case of complete synchrony, the numerator and denominator of R will be equal, as $X = x_i$, implies $R = 1$. For periodic orbits, $\langle X^2 \rangle = \langle X \rangle^2$

implies $R = 0$ and $R \sim 0.5$ shows the absence of global coherence. We also applied this metric to quantify the pairwise magnitude order by choosing $N = 2$ not just for the whole system.

We show that the orbits generated from the logistic map achieve CS in Sec. 4.2.3, as the absolute difference between node 1 and 3 became exactly zero after a certain point (see Fig. 4.4). However, examining all pairs in a very large system can be challenging, and it is impossible to quantify the level of synchrony without a metric. Therefore, we use the order parameter R to quantify the magnitude order of all systems in Fig. 4.11(a), (b), respectively. The magnitude order, R, is approximately 0.4, which is evidence that there is no global coherence in the whole system; however, we can see that some pairs are highly synchronised in Fig. 4.11(b) with the highest value between nodes 1 and 3 as greater than 0.8.

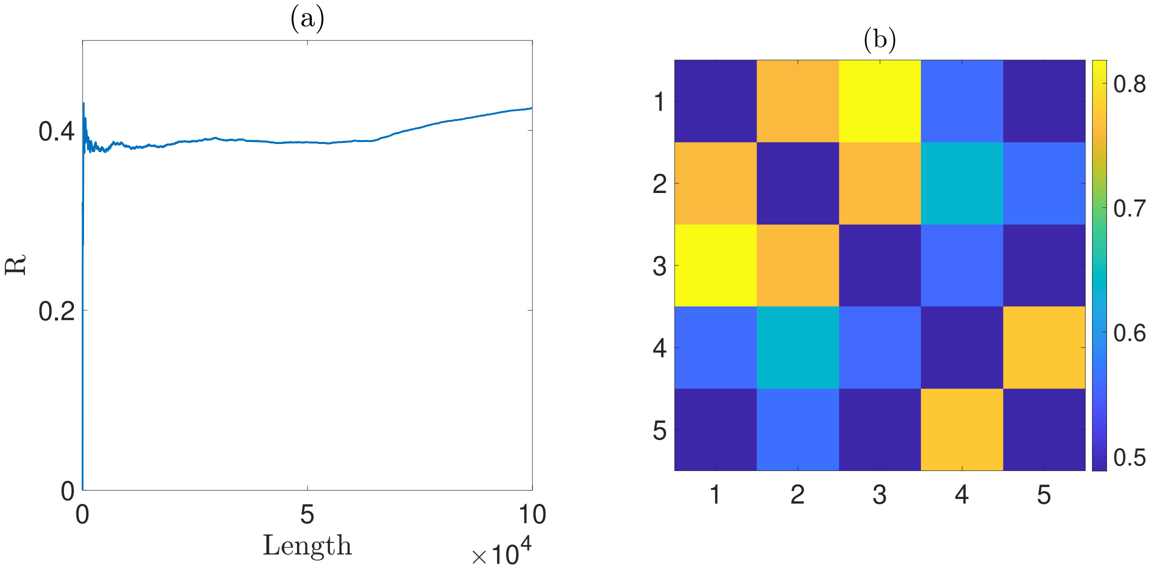

Figure 4.11: **Amplitude order of an example of logistic map**. (a) The amplitude order of the whole system as a function of time. (b) Pairwise amplitude order with the highest one between 1-3, since CS occurred between them after 6×10^4 iterations as shown in Fig. 4.4.

4.4.2 Kuramoto Phase Order

Kuramoto model is a well-known phase-coupled model to investigate the synchronisation behaviour of the oscillators as given by the equation. 3.17. Kuramoto order parameter, ρ, measuring the synchronisation level of phase oscillators, was introduced in [37] as follows:

$$\rho(t) = \left| \frac{\frac{1}{N}\sum_{j=1}^{N} e^{i\theta_j(t)}}{e^{i\Psi}} \right|, \tag{4.14}$$

where Ψ is the mean of the phases over oscillators on time, N is the number of oscillators and $\theta_j(t)$ is the phase of the j^{th} oscillator at time t, $|\cdot|$ denotes the modulus of a complex number.

4.4.3 Pairwise Phase Order

We use the pairwise phase order as a metric to quantify pairwise phase order by following [27], which is,

$$C_{i,j}(t) = \lim_{\Delta t \to \infty} \left| \frac{1}{\Delta t} \int_{\tau}^{\tau+\Delta t} e^{i[\theta_i - \theta_j]} d\tau \right|, \tag{4.15}$$

where Δt is time period; which is $t_f = 10^4$ in Rössler system discussed in Sec. 4.3.4 as we consider the entire duration of time, θ_i denotes the phase of the i^{th} node.

Although some pairs in the Rössler system achieve phase synchronisation (see Fig. 4.10), the Kuramoto phase order, ρ, remains relatively small at 0.3, as depicted in Fig. 4.12(a). We examined pairwise relations in terms of both PS and PC of the trajectories, comparing them with those of TWSD generated from the original data. This process destroys the phase relation while recovering the linear correlation between pairs of trajectories by randomising the Fourier phases. Additional details can be found in Sec. 6.1.2.2. Figure 4.12(b) shows the pairwise phase order of the original phases and TWSD phases. The phase order of synchronised pairs; 1-2, 1-3, 2-3, and 4-5 (see panel (a) in 4.10), is higher than 0.8. This shows that it allows us to distinguish synchronised pairs from the others. The pairwise phase orders in TWSD are very close to zero because it is obtained from the original data, destroying the entire phase relation using the Fourier transform. Figure

4.12(c) demonstrates the PC between the pairs in the original data and in the TWSD; the PC values are exactly the same in both, since TWSD recovers the linear relationship in the original data. Figure 4.12(d) shows the corresponding p-values of PC, and PC among all pairs are statistically significant at the level of significance 0.05.

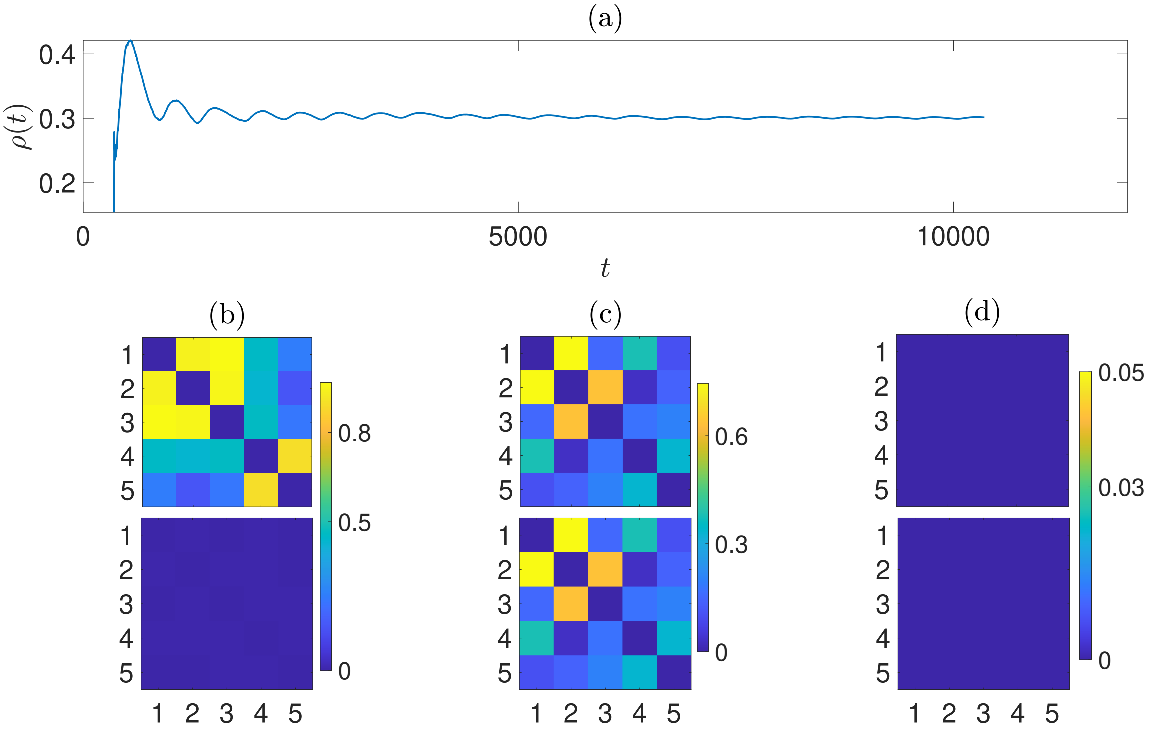

Figure 4.12: **Phase order of an example of Rössler system** (a) Kuramoto order, ρ, of the entire system as a function of time. (b) Pairwise phase order of original data and its TWSD (c) PC between the pairs in original data and its TWSD and (d) corresponding p-values.

4.5 Summary

At first glance, chaotic synchronisation may seem controversial, as chaotic systems are highly sensitive to initial conditions, and synchrony implies moving together in harmony. However, synchronisation of amplitudes or phases can be observed under certain coupling configurations. In this section, we present an example of CS between a pair of nodes of a trajectory iterated from the logistic map. We also discussed examples of PS and IPS between pairs of trajectories from the Rössler system. To quantify amplitude and phase order, we introduce three measures from the literature because understanding the behaviour of a system with many components is essential. We will revisit these examples in Chapter 5 to investigate their effects on network inference.

In addition, we delved into frequency analysis of signals ranging from periodic motion to more complex ones. The Fourier transform is a well-known method for decomposing signals into their sine-wave components and identifying dominant frequencies. It also enables noise reduction, as it is robust to noise when identifying dominant frequencies. However, we demonstrate that it cannot provide information for frequency-varying signals, which are examples of nonstationary signals. As we study network inference methods using information-theoretic approaches with chaotic data, we will determine instantaneous amplitudes, phases, and frequencies using the concept of the analytic signal with the Hilbert Transform (HT). We will use the FT to generate TWSD in Sec. 6.1.2.2, and the HT in the concept of amplitude-phase modulated surrogate data in Chapter 7.

Chapter 5

Information-theoretical Approaches

5.1 Motivation

To assess the similarity between two time series, PC is a commonly used method because its interpretation is quite straightforward. A positive correlation implies that one variable is directly proportional to another, while a negative correlation implies that one is inversely proportional to another. Despite its convenience in obtaining a preliminary understanding of the data, PC has limitations in capturing nonlinear relationships among variables. However, data from various fields, ranging from biology to finance, often exhibit nonlinear relationships. Information theory has emerged as a solution to this challenge. In network inference from time series data, information theoretical approaches are commonly used to consider not only linear relation but also nonlinear relations between nodes. From this perspective, information-based theoretical approaches are superior to PC as shown in a simple example of the network of 4 nodes where nodes 1 and 2 linearly interact to each other and nodes 3 and 4 interact non-linearly as seen in Fig. 5.1. As data generation process, 10^4 uniformly random generated points in $[0, 1]$ have been assigned to node 1 and 3 to create data x_1 and x_3, respectively. This results in a weak (or almost no) correlation between x_1

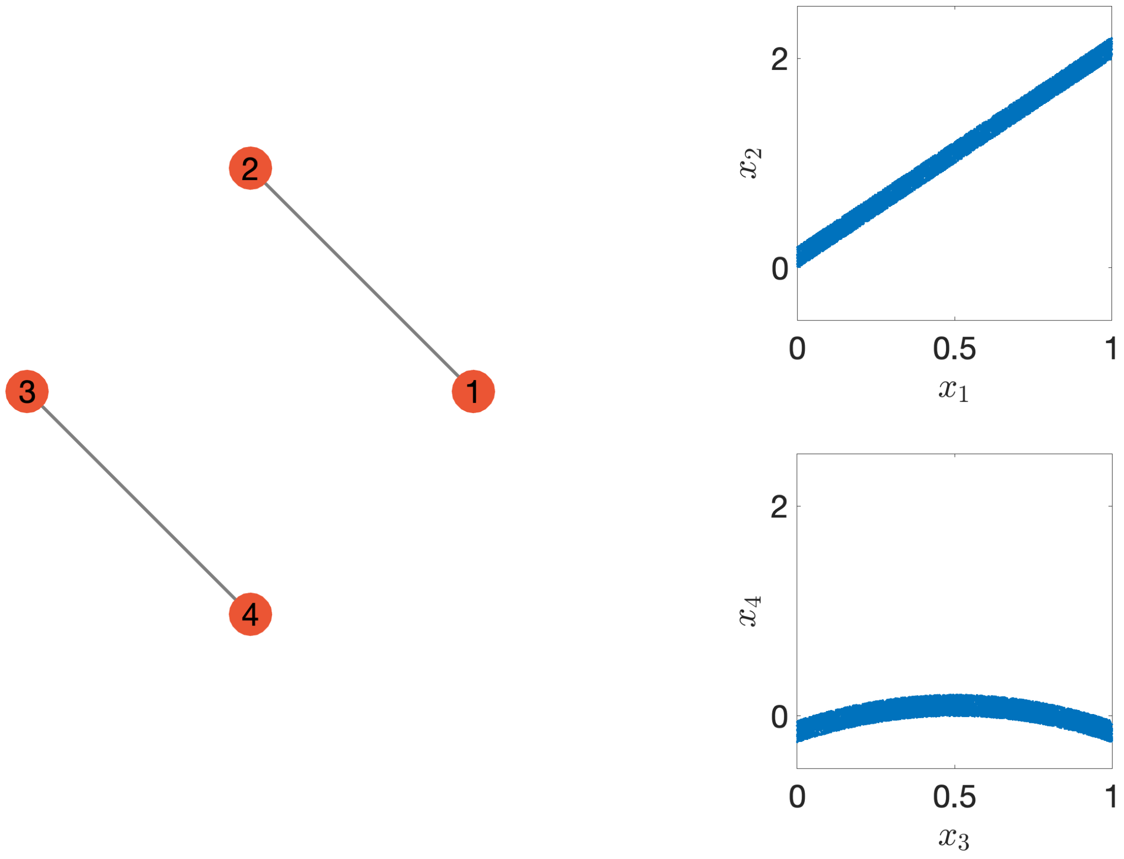

Figure 5.1: **Linear and nonlinear relation.** (a) Network of 4 nodes, nodes 1 and 2 interact linearly, nodes 3 and 4 interact non-linearly. (b) Linear trend between nodes 1 and 2. (c) Nonlinear relation between nodes 3 and 4.

and x_3. The data for node 2 has been calculated from the data of node 1 as $x_2 = 2x_1 + 0.2\xi$ and for node 4 calculated from the data of node 3 as $x_4 = -(0.5 - x_3)^2 + 0.2\xi$, where ξ is a uniformly random number in $[0, 1]$ to account for noise in the system. Eventually, the structure of the network among the nodes is shown in Fig. 5.1(a). As expected due to the data generation process, x_1 and x_2 follow a linear trend as shown in Fig. 5.1(b) while x_3 and x_4 accumulate around a parabolic curve in Fig. 5.2(c).

Pearson correlation and mutual information (MI) are implemented on the data and bar plots demonstrate that PC among the pairs on panel (a) and MI on panel (b) in Fig. 5.1. It results in that PC only captures the linear interaction between x_1 and x_2, whereas MI captures both linear and nonlinear interaction between nodes 3 and 4. Thereby, we

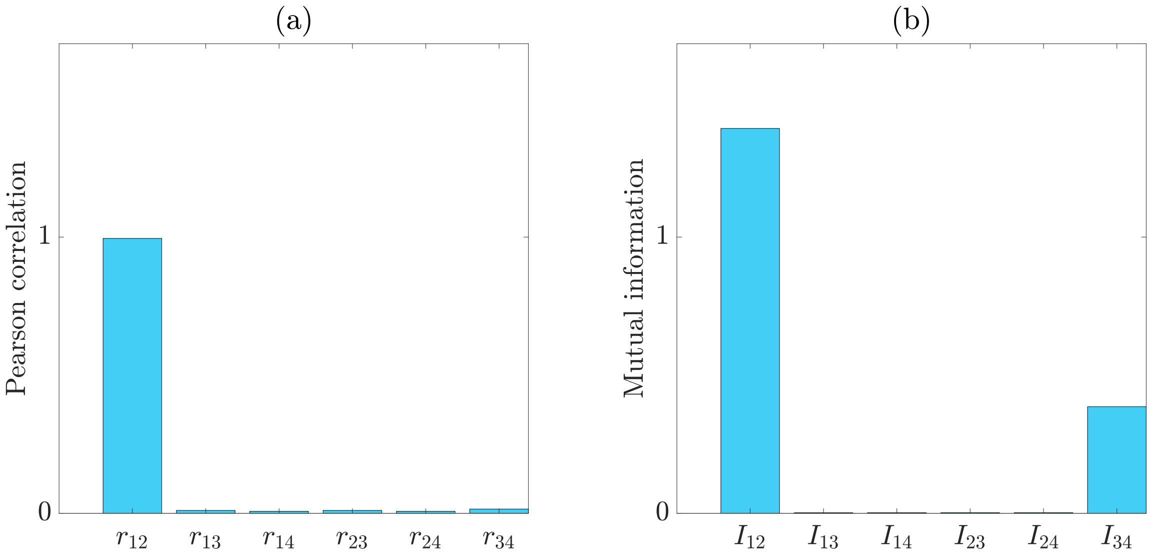

Figure 5.2: **Pearson correlation vs mutual information.** (a) Pearson correlation captures only the linear relation between nodes 1 and 2. (b) Mutual information captures not only linear but also nonlinear interactions between nodes 3 and 4. Note that Pearson correlation and mutual information of other pairs are very small numbers, so they are barely seen in the plot.

prefer to use information theoretical approaches in network inference to capture nonlinear interdependencies.

In this chapter, we present the theoretical framework for information-based approaches and their implementations on different types of data, including Gaussian distributed correlated data, coupled logistic and circle maps, coupled HR model and coupled Lorenz system. Eventually, we present MIRs among the nodes in two systems, coupled logistic and coupled Rössler, that we observed synchronisation in Chapter 4.

5.2 Information Theory

5.2.1 Shannon Entropy

In 1948, Claude Shannon published a groundbreaking work on Communication Theory [59], which laid the foundation for modern communication devices and systems. At that time, communication methods were relatively primitive and controlling noise in the communication process was a significant challenge. Shannon introduced the concept of a binary digit (or ”bit”) and used Markov processes to model the probabilities of events based on previous events. He introduced a measure to quantify the amount of information transferring among the systems using the concept of entropy and formalised it. Here, we will focus mainly on the concepts of entropy, MI, and MIR, which are the fundamental concepts in various fields, including information theory, statistics, and signal processing.

Definition 5.2.1 (Shannon Entropy)**.** *The Shannon entropy, which measures the amount of uncertainty in a random event, X is defined by*

$$\begin{aligned} H_X &= -\sum_{x \in \mathcal{X}} p_x \log p_x \\ &= -\mathbb{E}\big[\log p_x\big], \end{aligned} \tag{5.1}$$

where x is an event that belongs to the set of all possible events, denoted by $\mathcal{X}$ and $p(x)$ is the probability of x to occur. Once the base of the logarithm is 2, the unit of entropy is bits.

To gain a better understanding of the definition above, we focus on a simple example of a coin flip. Let $\mathcal{X}$ be a set of 2 possible outcomes: heads and tails. The probabilities of the outcomes are p (or $1 - p$) with heads (or tails) occurring. The probability mass function of the events,

$$p_x = \begin{cases} p, & \text{when heads occur} \\ 1 - p, & \text{when tails occur} \end{cases}.$$

The entropy as a function of p can be derived from Eq. (5.1) and is given by

$$H(p) = -p \log p - (1 - p) \log(1 - p).$$

Figure 5.3(a) demonstrates the relation between probability and entropy, where the entropy reaches its maximum value when probability $p = \frac{1}{2}$, which means the maximum uncertainty for a fair coin. In order to ensure continuity, we assume $p \log(p) \to 0$ as $p \to 0$. The amount of uncertainty in the case of two events X and Y occurring together that is the joint entropy of X and Y, can be easily defined using the fact that $H_X = -\mathbb{E}[\log p_x]$,

$$H_{XY} = -\mathbb{E}[\log p_{xy}] = -\sum_{x\in X}\sum_{y\in Y} p_{xy}\log p_{xy}, \tag{5.2}$$

where p_x and p_y are the marginal probabilities of the variables and p_{xy} is the joint probability.

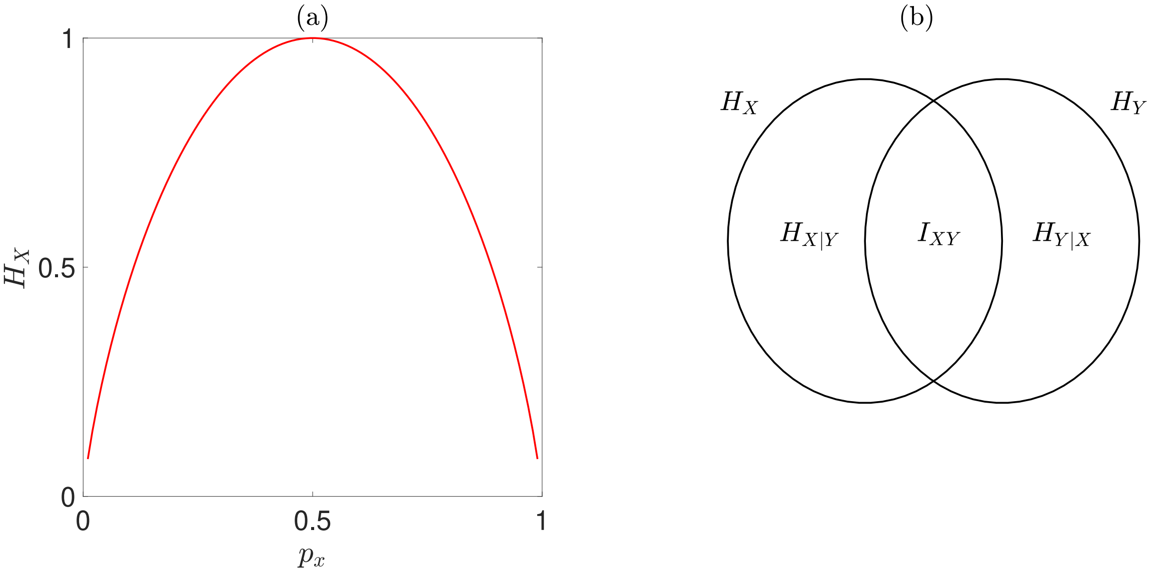

Figure 5.3: **Relation among probability, entropy and MI.** (a) Entropy as a function of probability p_x for the coin flip. (b) The scheme shows the relation between MI and marginal and conditional entropy.

Lemma 5.2.1. *The joint entropy between two events X and Y is symmetric*

$$H_{XY} = H_{YX}.$$

Proof.

$$\begin{aligned} H_{XY} &= \sum_{x\in\mathcal{X},y\in\mathcal{Y}} p_{xy}\log p_{xy} \\ &= \sum_{x\in\mathcal{X},y\in\mathcal{Y}} p_{yx}\log p_{yx} && \text{(where } p_{xy} = p_{yx}\text{)} \\ &= H_{YX}. \end{aligned}$$

$\square$

Definition 5.2.2 (Conditional Entropy)**.** *Let* p_x *and* p_y *be the marginal probabilities,* p_{xy} *is joint and* $p_{y|x}$ *is the conditional probability of two discrete events* X *and* Y*. The conditional entropy of* Y *given* X*,*

$$\begin{aligned} H_{Y|X} &= -\sum_{x\in\mathcal{X}} p_x H_{Y|X=x} \\ &= -\sum_{x\in\mathcal{X}} p_x \sum_{y\in\mathcal{Y}} p_{y|x}\log p_{y|x} && \textit{(from Bayes' rule)} \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log p_{y|x}. \end{aligned} \tag{5.3}$$

Theorem 5.2.2 (Chain Rule)**.** *The joint entropy of random events* X *and* Y *is equal to marginal entropy of* X *and conditional entropy of* Y *given* X *as a result of Bayes' rule, which is given by,*

$$H_{XY} = H_X + H_{Y|X}. \tag{5.4}$$

Proof.

$$\begin{aligned} H_{XY} &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log p_{xy} && \text{(from Eq. 5.2)} \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log\left(p_x p_{y|x}\right) \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}\log p_x - \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}p(x|y) && \text{(from Bayes' rule)} \\ &= -\sum_{x\in\mathcal{X}} p_x\log p_x - \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p_{xy}p(x|y) && \text{(from Eqs. 5.1, 5.3)} \end{aligned}$$

Hence $H_{XY} = H_X + H_{Y|X}$.

$\square$

5.2.2 Mutual Information

Definition 5.2.3. *Kullback-Leibler distance (or the relative entropy) between two probability mass function, p_x and q_x is given by*

$$D(p||q) = \sum_{x \in \mathcal{X}} p_x \log \frac{p_x}{q_x}. \tag{5.5}$$

Definition 5.2.4. *Let us suppose that p_x and p_y are marginal probabilities and p_{xy} is the joint probability for two random variables X and Y. Mutual information, I_{XY}, is the Kullback-Leibler distance between the joint probability p_{xy} and the multiplication of marginal probabilities, $p_x p_y$, can be written as*

$$\begin{aligned} I_{XY} &= D(p_{xy}||p_x p_y) \\ &= \sum_{x \in X} \sum_{y \in Y} p_{xy} \log \frac{p_{xy}}{p_x p_y}. \end{aligned} \tag{5.6}$$

From Eq. (5.6), if two random variables X and Y are independent, the joint probability of them is equal to the multiplication of marginal probabilities, $p_{xy} = p_x p_y$. This implies that MI of two independent variables is equal to zero.

Lemma 5.2.3. *MI between the random variables X and Y can be interpreted as the reduction in uncertainty of X due to knowledge of Y.*

Proof.

$$\begin{aligned} I_{XY} &= \sum_{x \in X} \sum_{y \in Y} p_{xy} \log \frac{p(x|y)}{p_x} \\ &= - \sum_{x \in X} \sum_{y \in Y} p_{xy} \log p_x - \left(- \sum_{x \in X} \sum_{y \in Y} p_{xy} log p(x|y) \right) \\ &= - \sum_{x \in X} p_x \log p_x - \left(- \sum_{x \in X} \sum_{y \in Y} p_{xy} \log p(x|y) \right). \end{aligned}$$

Hence $I_{XY} = H_X - H_{X|Y}$.

□

Lemma 5.2.4. *Mutual information is a symmetric measure.*

$$I_{XY} = I_{YX}.$$

Proof. We know that

$$I_{XY} = H_X - H_{X|Y}.$$

From Eq. (5.2.2), if we substitute $H_X = H_{XY} - H_{Y|X}$ into the previous equation, we obtain

$$\begin{aligned} I_{XY} &= H_{XY} - H_{Y|X} - H_{X|Y} \\ &= H_{YX} - H_{X|Y} - H_{Y|X} && (H_{XY} = H_{YX} \text{ from Lemma 5.2.1}) \\ &= I_{YX}. \end{aligned}$$

□

In information theory, the concavity of a function is important as it allows showing some properties of the mutual information, discussed later in this section. The definition of a concave function is given in the following and is visualised in Fig. 5.4.

Definition 5.2.5 (Concave function)**.** *Let $I \subset \mathbb{R}$ be an interval. A function $f : I \to \mathbb{R}$ is a concave function if for every $x_1, x_2 \in I$ and $t \in [0, 1]$ we have*

$$tf(x_1) + (1-t)f(x_2) \leq f(tx_1 + (1-t)x_2). \tag{5.7}$$

Lemma 5.2.5 (Jensen's inequality)**.** *For a discrete event X, its probability mass function with k mass points, $p_1, p_2, \ldots, p_k$. The function f is concave and its expected value is denoted by $\mathbb{E}f(X)$. The expected value of $f(X)$ is less than or equal to the expected value of X under the function f, $f(\mathbb{E}X)$, and is given by,*

$$\mathbb{E}f(X) \leq f(\mathbb{E}X). \tag{5.8}$$

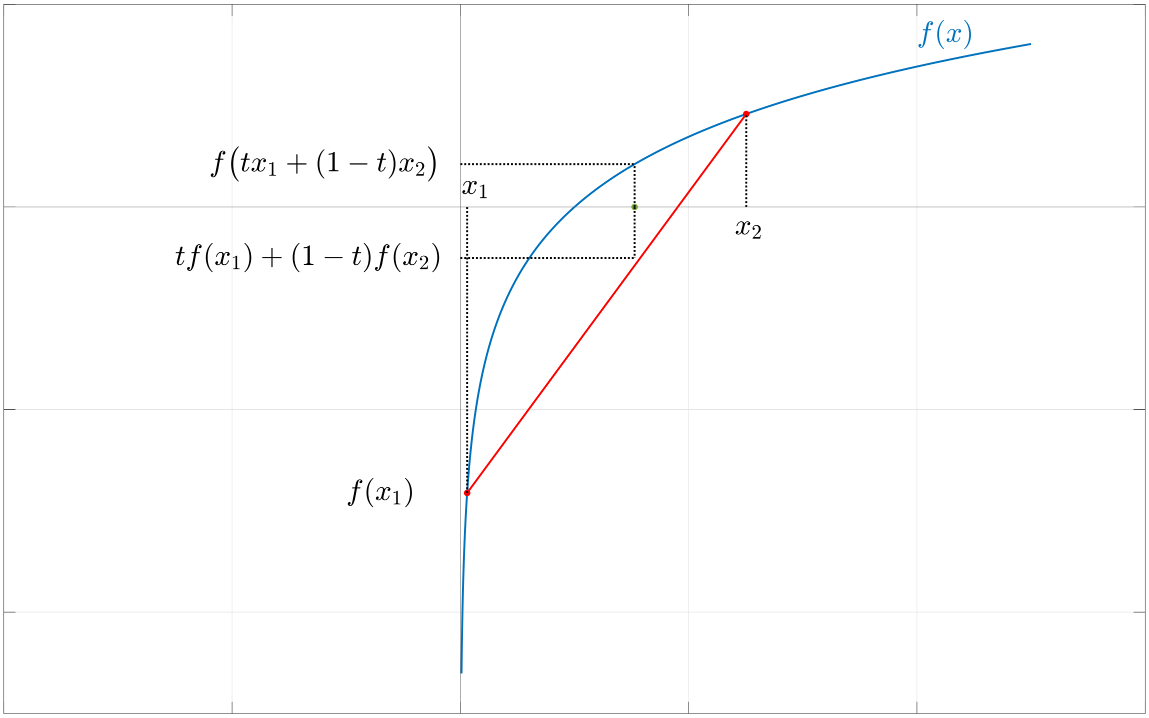

Figure 5.4: **Visualisation of the definition of a concave function.** Linear combination of the images of x_1 and x_2 under the function f is less than the image of linear combination of the points x_1 and x_2 under f, as given in Eq. (5.7).

Proof. From the definition of a concave function, we know that inequality (5.8) holds for two mass points with the equality, $p_2 = 1 - p_1$,

$$p_1 f(x_1) + p_2 f(x_2) \leq f(p_1 x_1 + p_2 x_2).$$

We can generalise the inequality to probability distributions with k mass points by using mathematical induction. Let us assume that the inequality holds for the probability distributions with $k-1$ mass points.

$$\sum_{i=1}^{k-1} p_i f(x_i) \leq f\left(\sum_{i=1}^{k-1} p_i x_i\right).$$

We want to show that the inequality holds for the probability distributions with k mass points. If we reconstruct the probability values as $p'_i = \frac{p_i}{1-p_k}, \forall i = 1, 2, \ldots, k-1$, then

$$\begin{aligned}\mathbb{E}f(X) = \sum_{i=1}^{k} p_i f(x_i) &= p_k f(x_k) + (1-p_k) \sum_{i=1}^{k-1} p'_i f(x_i) \\ &\leq p_k f(x_k) + (1-p_k) f\Big(\sum_{i=1}^{k-1} p'_i x_i\Big) \qquad (5.9) \\ &\leq f\Big(\sum_{i=1}^{k} p_i x_i\Big) = f(\mathbb{E}X). \qquad (5.10)\end{aligned}$$

The first inequality in Eq. (5.9) follows the induction hypothesis and the second inequality in Eq. (5.10) follows the definition of a concave function. □

Lemma 5.2.6. *The relative entropy is non-negative with the equality if the random variables X and Y associated with probability mass functions, p_x and q_x, are independent.*

$$D(p||q) \geq 0. \qquad (5.11)$$

Proof.

$$\begin{aligned}-D(p||q) &= -\sum_{x\in\mathcal{X}} p_x \log \frac{p_x}{q_x} \\ &= \sum_{x\in\mathcal{X}} p_x \log \frac{q_x}{p_x} \\ &\leq \log \sum_{x\in\mathcal{X}} p_x \frac{q_x}{p_x} \qquad (5.12) \\ &= \log \sum_{x\in\mathcal{X}} q_x = \log 1 = 0. \qquad (5.13)\end{aligned}$$

Hence, $D(p||q) \geq 0$. Here, in inequality (5.12) is a result of Jensen's inequality (5.8) as log is a concave function. □

Lemma 5.2.7. *Mutual information is a non-negative measure with equality if and only if X and Y independent.*

$$I_{XY} \geq 0.$$

Proof. The mutual information, $I_{XY} = D(p_{xy}||p_x p_y)$ and Kullback-Leibler distance between the joint entropy and the multiplication of marginal probabilities,

$$D(p_{xy}||p_x p_y) \geq 0,$$

as shown in Lemma (5.2.6), hence, $I_{XY} \geq 0$ with the equality if and only if the events X and Y are independent. □

5.2.2.1 Probability Estimation from Time-series Data

When the probability mass function is unknown for a given time series data, there are three methods to compute the probabilities involved in Eq. (5.6), namely: (a) the bin or histogram method [20], (b) the kernel density method [28, 43] and (c) the method of estimating probabilities from the distance between closest neighbours [36]. All methods have their own pros and cons, but we use the bin method to calculate the probabilities in Eq. (5.6) using equally-sized cells in a partition of the probabilistic space Ω of X and Y, following [2,12,26] for two reasons: (1) The binning method is straightforward to implement and interpret the results. (2) We will use information-based approaches to compare pairs in a system of N nodes. Since the number of pairs, $\frac{N(N-1)}{2}$, increases exponentially, we prefer a method that maintains a balance between computational cost and precise results.

To compute the probabilities in (5.6), we start by assuming that we have the time series of two random variables X and Y of equal length L. Then, the probabilistic space Ω can be partitioned into $N \times N$ equally sized cells of size $\epsilon = 1/N$. A 5×5 partition of equally sized cells of two random variables, X and Y drawn from the uniform distribution, is shown in Fig. 5.5, where X and Y are both translated into the interval $[0, 1]$, to avoid numerical roundoff errors in the computation of probabilities. For this reason, in the following, we will be translating all time series in this interval and calculate the marginal and joint probabilities based on the following definitions

$$P_{XY}(i,j) = \frac{L_{XY}(i,j)}{L}, \quad P_Y(i) = \frac{L_Y(i)}{L}, \quad P_X(j) = \frac{L_X(j)}{L}. \tag{5.14}$$

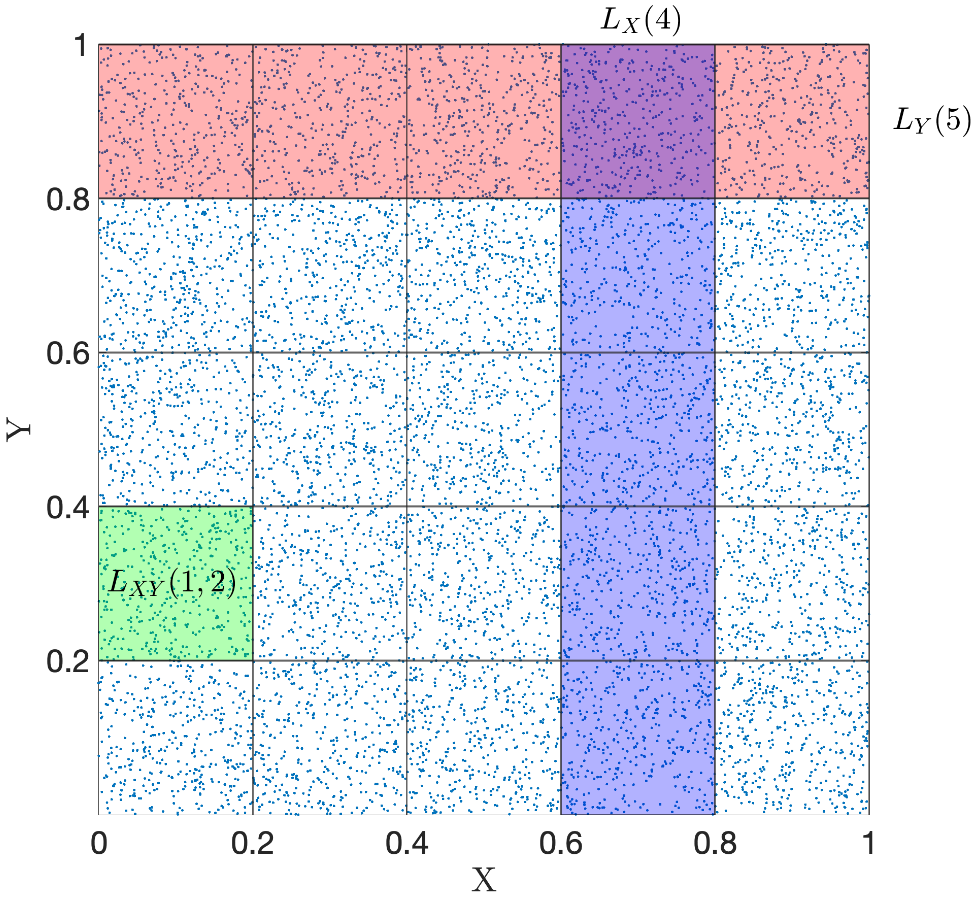

Figure 5.5: **The binning method.** Plot of time series X and Y in the plane $[0,1] \times [0,1]$ and a partition of $N \times N = 5 \times 5 = 25$ cells of equal size $\epsilon = \frac{1}{N}$. The notations $L_X(4)$, $L_Y(5)$ and $L_{XY}(1,2)$ denote the number of points in the purple column, red row, and green cell, respectively.

Here, L is the length of the time series, $L_{XY}(i,j)$ is the number of points in cell (i,j), $L_Y(i)$ is the number of points lying in the i^{th} row and $L_X(j)$ is the number of points lying in the j^{th} column, where $i, j = 1, 2, \ldots, N$. In this context, I_{XY} depends on the number of bins, N, to partition Ω into N^2 equally-sized cells. Hence, different partition sizes produce different probability values, thus the probabilities in Eq. (5.14) depend on N. Consequently, I_{XY} depends on N, which we denote by $I_{XY}(N)$. We follow [2,12,26], to calculate the average I_{XY} across partitions for increasing N by considering all N values that satisfy

$$\frac{L}{N^2} \geq N_{oc}, \tag{5.15}$$

where N_{oc} is the total number of occupied cells. Hence, Eq. (5.15) gives the maximum partition size, N_{max}, which guarantees the computations of the probabilities in Eq. (5.14) are not affected by a large number of poorly occupied or empty cells. By poorly occupied cells, we mean cells with less than N_{oc} data points.

5.2.3 Mutual Information Rate

Mutual information rate can be defined as the amount of information transferred between two random variables, X and Y with equal length L, per unit of time,

$$\mathrm{MIR}_{XY}(N) = \lim_{L\to\infty} \frac{I_{XY}(N)}{L} \tag{5.16}$$

In memoryless systems, such as chaotic systems, pairs of variables lose their correlation after a CDT, T, meaning they become unpredictable after this time. The authors in [3] have shown that, in the case of finite length, L, and partition size, N, MIR can be approximated by

$$\mathrm{MIR}_{XY}(N) \approx \frac{I_{XY}(N)}{T(N)}, \tag{5.17}$$

where $T(N)$ is the CDT of pair X, Y in Ω, partitioned in an $N \times N$ grid. Following [2,12,26], we compute $I_{XY}(N)$, $T(N)$ for all N that satisfy Eq. (5.15). This gives the range of N values, $\tilde{N} = (N_{min}, N_{max})$, where $I_{XY}(N)$, $T(N)$ are computed. In our work, we consider $N_{min} = 0.1N_{max}$ to guarantee N_{min} is small enough compared to N_{max}.

The data points of chaotic systems within a closed region spread over the whole probability space, Ω, after correlation decay time, T. Following [12], we used the itinerary network model to estimate CDT. In an $N \times N$ partition of Ω, each cell is regarded as a node in an itinerary network, $\mathcal{G}_N$, given by the $N^2 \times N^2$ adjacency matrix $\mathbf{G}(N) = \{G_{ij}(N)\} = \{0, 1\}$, where $i, j = 1, \ldots, N^2$. Thus, an entry of 1 in $\mathbf{G}(N)$ corresponds to the case where the points in cell i move to cell j, and 0 when they do not. Consequently, $T(N)$ can be defined as the diameter of $\mathcal{G}_N$ as $T(N)$ is the minimum time it takes for the points in any cell of a partition in Ω to spread to the whole extent of Ω. By definition, the diameter of a network is the maximum length for all shortest-paths (see the discussion in Sec. 2.2.4), i.e., the minimum distance required to cross the entire network. This approach transforms the calculation of $T(N)$ into the calculation of the diameter of $\mathcal{G}_N$. In particular, we use the built-in Matlab function `distances` to compute all shortest-path lengths between pairs in $\mathcal{G}_N$ and from those, the diameter of $\mathcal{G}_N$ as the maximum of all shortest-path lengths.

For a pair X, Y, we compute $\mathrm{MIR}_{XY}(N)$ using Eq. (5.17) for all N in $\tilde{N}$, and from that, MIR_{XY} as the average of $\mathrm{MIR}_{XY}(N)$ over N in $\tilde{N}$. In a network of M nodes, there are $\frac{M(M-1)}{2}$ unique pairs X, Y, excluding self-connections and connections Y, X as $\mathrm{MIR}_{XY} = \mathrm{MIR}_{YX}$, i.e. the MIR matrix is symmetric. This results in saving computational time as one has to compute $\frac{M(M-1)}{2}$ of X, Y pairs instead of M^2 pairs.

5.2.4 Other Methods

To measure similarity between pairs of nodes X and Y, we also use the Pearson correlation, double normalised MIR($\overline{\mathrm{MIR}}$), lagged-PC and lagged double normalised MIR.

5.2.4.1 Double Normalised Mutual Information Rate

As different systems may have significantly different CDTs, resulting in MIR values in different scales, it becomes necessary to standardise the MIR measure within the interval $[0, 1]$. To achieve this standardisation, the authors in [12] propose the double normalised MIR, denoted by $\overline{\mathrm{MIR}}$. The normalisation steps are as follows

1. **Linear scaling:** MIR values obtained in different partitions are firstly scaled to $[0, 1]$ by preserving the proportion of the distance between pairs.

$$\widehat{\mathrm{MIR}}_{XY}(N) = \frac{\mathrm{MIR}_{XY}(N) - \min\{\mathrm{MIR}_{XY}(N)\}}{\max\{\mathrm{MIR}_{XY}(N)\} - \min\{\mathrm{MIR}_{XY}(N)\}}.$$

2. **Second normalisation over different partitions:** It allows us to obtain only one MIR value per pair, with the equal effect of different partitions on final result

$$\overline{\mathrm{MIR}}_{XY} = \frac{\sum_i \widehat{\mathrm{MIR}}_{XY}(N_i)}{\max\{\sum_i \widehat{\mathrm{MIR}}_{XY}(N_i)\}}.$$

5.2.4.2 Pearson correlation

As we discussed earlier, PC captures the linear relation but not nonlinear ones. To compare its results with information-theoretic approaches, we also consider the absolute value of PC between two time series $X = x_1, x_2, \ldots x_L$ and $Y = y_1, y_2, \ldots, y_L$ of equal length L,

$$\mathrm{R}_{XY} = \frac{\sum_{i=1}^{L}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{L}\sqrt{(x_i - \overline{x})^2(y_i - \overline{y})^2}},$$

where $\overline{x}$ is the mean of vector x. In the context of network inference, we use the absolute value of PC thereafter.

5.2.4.3 Lagged-PC and Lagged-$\overline{\mathbf{MIR}}$

The Lagged-PC method considers two time series X and Y with the same length L at different lags, τ. We can reconstruct the time series based on the time lag, τ, as follows

$$\begin{aligned} X_\tau &= \{x_1, x_2, \ldots, x_{L-|\tau|}\} \quad \& \quad Y_\tau = \{y_{|\tau|+1}, y_{|\tau|+2}, \ldots, y_L\}, \qquad \tau \leq 0 \\ X_\tau &= \{x_{\tau+1}, x_{\tau+2}, \ldots, x_L\} \quad \& \quad Y_\tau = \{y_1, y_2, \ldots, y_{L-\tau}\}, \qquad \tau > 0. \end{aligned}$$

From the reconstructed X_τ and Y_τ, we compute the set of lagged-PC values, $\widetilde{\mathrm{PC}}_\tau = \{\mathrm{PC}_{X_\tau Y_\tau},\ \text{for all } \tau\} = \{\mathrm{PC}_{-\tau}, \mathrm{PC}_{-\tau+1}, \ldots, \mathrm{PC}_\tau\}$. Then we select the maximum value from the elements in the set of $\widetilde{\mathrm{PC}}_\tau$.

5.2.4.4 Lagged double normalised MIR

After reconstructing the time series X_τ and Y_τ in the same way, we compute the set of τ-lagged double normalised MIR, $\widetilde{\overline{\mathrm{MIR}}}_\tau = \{\overline{\mathrm{MIR}}_{X_\tau Y_\tau},\ \text{for all } \tau\}$. We select the maximum value from the elements of the set, $\widetilde{\overline{\mathrm{MIR}}}_\tau$, considering the time lags from $-\tau$ to τ.

5.3 Applications

In this section, we will discuss the results obtained from discrete, continuous and stochastic systems using different similarity measures including MI, MIR, double normalised MIR, PC, lagged-PC, lagged-double normalised MIR. The results are presented in 4 subsections as follows

1. We implemented PC and MIR on the data generated from multivariate Gaussian distributed correlated data based on a given covariance matrix.
2. We used PC, MI, double normalised MIR, lagged-double normalised MIR and lagged-PC on the data generated from two discrete maps, namely logistic and circle.
3. We used PC, MI, double normalised MIR, lagged-double normalised MIR and lagged-PC on the data generated from continuous systems, namely, HR and Lorenz.
4. To examine the effect of synchronisation on network inference, we implemented MIR on the data exhibiting CS, PS and IPS discussed in Secs. 4.2.3 and 4.3.4.

In order to measure the efficiency of the methods, we used the true positive rate (TPR) and false positive rate (FPR), calculated by

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \quad \& \quad \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}}, \tag{5.18}$$

where TP represents the number of correctly inferred links that are present in the original network, FP the number of erroneously inferred links that is actually non-exist in the original network, FN the number of missing links that is present in the original network, and TN the number of correctly identified non-existing links.

		Inferred Network	
		1	0
Original	1	TP	FN
	0	FP	TN

Table 5.1: **Confusion Matrix to compute TPR and FPR.** TP is the number of links that exist in the original and inferred networks, FP is the number of links inferred but not existing in the original network, FN is the number of links missed but existing in the original network, and TN is the number of links not inferred and not existing in the original network.

We also used the "ROC distance" which refers to the Euclidean distance between the point of (TPR, FPR) of an inference and the point of perfect inference, $(1, 0)$ in the plane of TPR versus FPR, the so-called ROC plane. It can be computed by,

$$\text{ROC distance} = \sqrt{(\text{TPR} - 1)^2 + \text{FPR}^2}. \tag{5.19}$$

5.3.1 Gaussian Distributed Correlated Data

We generate the data from a multivariate Gaussian distribution with a given mean vector and covariance matrix, following d-dimensional probability distribution function,

$$f(x, \mu, \Sigma) = \frac{1}{\sqrt{|\Sigma|(2\pi)^d}} e^{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)^T}, \tag{5.20}$$

where d is the number of variables, μ the mean vector of variables. In our case, we considered $d = 9$ and $\mu = 0$. The superscript T stands for the transpose of the related vector. Covariance matrix,

$$\Sigma = \begin{pmatrix} 3.40 & -2.75 & -2.00 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2.75 & 5.50 & 1.50 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2.00 & 1.50 & 1.25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.00 & 0.5 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & 0.3 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4.40 & -2.75 & -2.00 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2.75 & 5.50 & -1.00 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2.00 & -1.00 & 3.25 \end{pmatrix}.$$

As is well-known, the correlation between variables X and Y, denoted as R_{XY}, can be found by the covariance between X and Y dividing by multiplication of the standard deviations of X and Y, given by

$$R_{XY} = \frac{\Sigma_{XY}}{\sqrt{\Sigma_X \Sigma_Y}}, \tag{5.21}$$

where Σ_{XY} is the covariance between X and Y, represented by the off-diagonal elements in Σ, Σ_X the variance of variable X, corresponding to the diagonal elements in Σ). Based on this, we can derive the correlation matrix,

$$R = \begin{pmatrix} 1 & -0.64 & -0.97 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.64 & 1 & 0.57 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.97 & 0.57 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0.71 & 0.55 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 1 & 0.77 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.55 & 0.77 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -0.56 & -0.53 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.56 & 1 & -0.24 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.52 & -0.24 & 1 \end{pmatrix}. \tag{5.22}$$

As discussed in [26], the dynamics of real-life cases such as currency exchange rates in financial markets is often not well known. In such cases, data can be generated to reflect correlations between pairs of variables based on historical data. Therefore, we generate the data from the covariance matrix Σ using the multivariate Gaussian distribution in Eq. (5.20) with 10^5 time length. Figure 5.6(a) shows the plots of x_i, x_j for all $i, j = 1, \ldots, 9$, which is consistent with the correlation matrix R in Eq. (5.22). For example, Fig. 5.6 shows that the pair x_1, x_3 is highly anti-correlated as the points form a cigar-shaped cloud, falling very close to a line with a negative slope, indicative of strong anti-correlation. This is supported by $R_{13} = -0.97$ in Eq. (5.22). An intermediate case of correlation can be seen in the case of pair x_2, x_3, for which its plot in Fig. 5.6 looks like a wider cloud of points compared to the plot of pair x_1, x_3 in the same figure. This is a case of weaker positive correlation and is further corroborated by its corresponding R value in Eq. (5.22), which

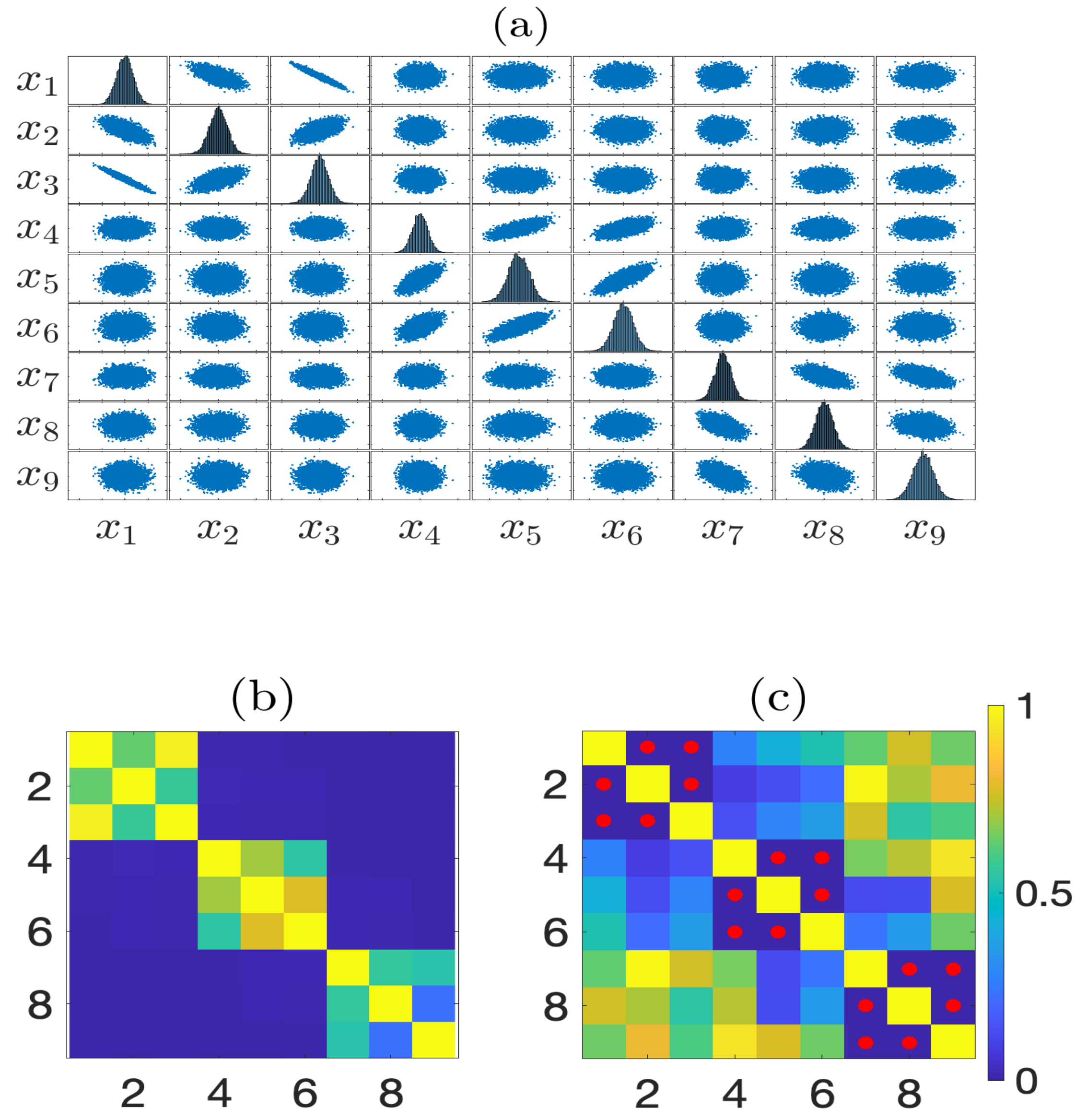

Figure 5.6: **Gaussian distributed correlated data**. (a) Plot of points of pairs x_i, x_j where $i, j = 1, \ldots, 9$, where highly correlated pairs are represented by cigar-shaped clouds of points, weakly correlated pairs by wider, cigar-shaped clouds of points and non-correlated ones by circular-like scattered points. The plots of nodes with themselves (along the diagonal of the figure) show the distribution of points of nodes x_i. Note that highly anti-correlated pairs are represented by cigar-shaped clouds of points with negative slope and that highly correlated pairs by cigar-shaped clouds of points with positive slope (b) The absolute values of PC are shown for the pairs. (c) Correspondingly, the p-values are shown with those below 0.05 indicated by red circles.

is 0.57. Finally, an interesting case arises for the pair x_8, x_9, as its plot in Fig. 5.6 shows an even wider cloud of points, indicating even weaker correlation among all pairs of nodes with non-zero correlation. In Fig. 5.6(b), the absolute value of the PC values computed from the data is shown by a colour map. Correspondingly, in Fig. 5.6(c), the p-values are shown, with red circles indicating pairs where the p-values are statistically significant, being less than 0.05.

By the assumption of that the pairs with $|R| > 0.5$, as we want to test the ability of the method to exclude weak correlation even if it is statistically significant with $p < 0.05$, the network structure has been shown in Fig. 5.7(a). The bar graph in Fig. 5.7(b) shows MIR values between pairs, with purple bars representing connected pairs and blue bars representing the unconnected pairs. By selecting a threshold within the grey shaded area, one can reconstruct the network structure. Similarly, Fig. 5.7(c) shows the results for PC. Therefore, both MIR and PC achieve to infer network topology for carefully chosen threshold values. This is evidence that MIR can capture the linear correlation between pairs in a network inference.

5.3.2 Deterministic Discrete Systems

From the model Eqs. 3.13, 3.6, and 3.12, the data have been generated from 0.03 coupled logistic and circle maps with the length of time series 10^5 using the 16 node network structure (see Fig. 3.10). We implement the methods of $\overline{\text{MIR}}$, PC, lagged-double normalised MIR and lagged PC on the data.

Given that we are analysing a chaotic logistic trajectory with a positive MLE, 0.57, we expect the correlation between pairs to decay over time. Hence, the lag range was restricted to a limited interval, as lagged versions of the methods select the maximum value across the lags, τ. Figure 5.8(a) shows that MI and MIR as a function of τ for a connected pair, and the vertical red-dashed line specifies the interval of $[-T, T]$, where T is CDT estimated from the itinerary network method. After the estimated CDT, the trajectory starts to lose correlation with the past states. Therefore, we consider time delays ranging from $\tau_{min} = -10$ to $\tau_{max} = 10$ in the lagged methods. The bar graph in Fig. 5.8(b) shows

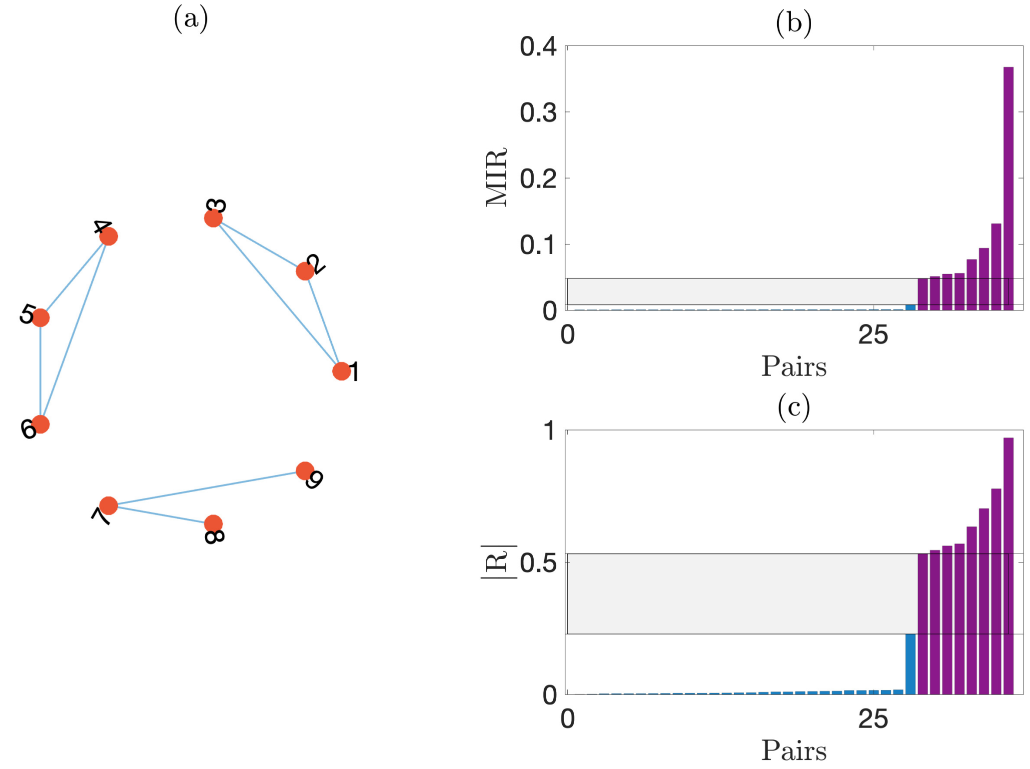

Figure 5.7: **Network inference from the Gaussian distributed correlated data**. (a) Network structure where links exist between pairs if $|R_{XY}| > 0.5$. (b) Network inference using MIR: purple bars represent connected pairs, blue bars represent unconnected pairs. The horizontal grey band indicates the threshold interval for perfect network inference. (c) Inference of the network using the absolute value of PC. Both methods show a range of threshold values for successful network inference.

$\overline{\text{MIR}}$ values between the pairs, where the purple bars are the connected pairs and the blue bars the unconnected pairs. For a threshold selected within the shaded area interval, one can reconstruct the network successfully. Figure 5.8(c) illustrates the same content for PC and there is no threshold to infer the correct network structure. Figures 5.8(d) and (e) visualise the results of lagged-doubled normalised MIR and lagged-PC for the time lags from $\tau_{min} = -10$ to $\tau_{max} = 10$ and both methods result in successful network inference. Consequently, the lagged versions of the methods improve the performance of network inference, especially in PC.

Similarly, Fig. 5.9 presents the results for the 0.03 coupled circle map whose trajectory shows chaotic dynamics with positive MLE, 1.32. In Fig. 5.9(a), MI and MIR as a function

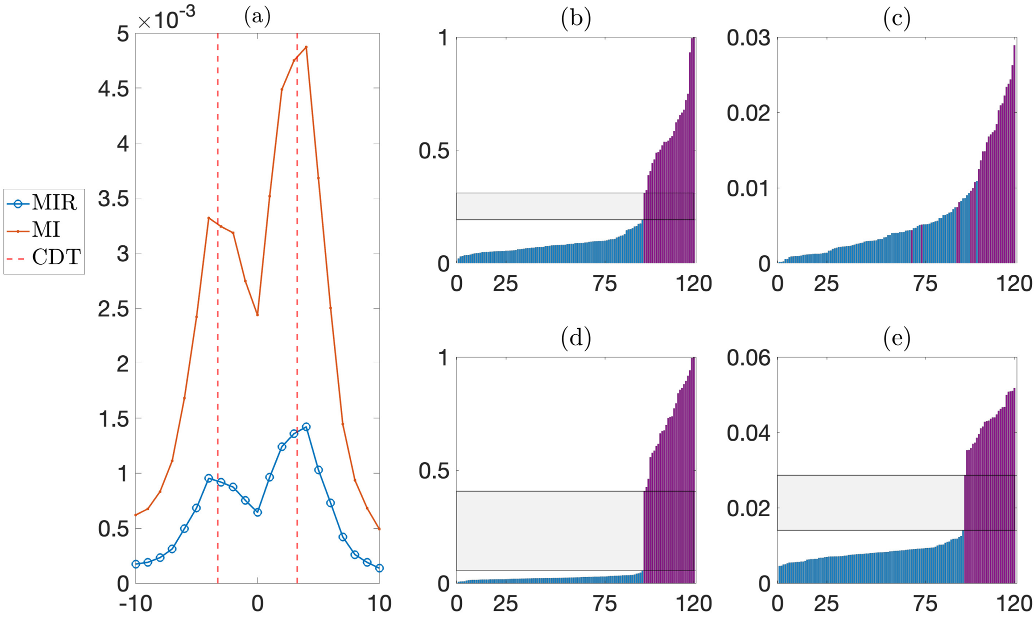

Figure 5.8: **Network inference from a system of coupled logistic maps**. (a) The values of MI and MIR from a connected pair of nodes are shown as functions of lag, τ. (b) Network inference using $\overline{\text{MIR}}$, (c) using PC, (d) using lagged-double normalised MIR and (d) using lagged-PC.

of lag, τ are shown, where the vertical red dashed line indicates the interval of $[-T, T]$. It is apparent that both MI and MIR decrease significantly after this interval, and lagged versions of the methods select the maximum value of MIR and PC. Therefore, we use the lagged versions of the methods in the lag interval $[\tau_{min}, \tau_{max}] = [-10, 10]$.

Since there exists a range of thresholds applicable in network inference, the double-normalised MIR method achieves successful network inference, as demonstrated in Fig. 5.9(b). In contrast, no threshold value allows for the reconstruction of the network using PC, as observed in Fig. 5.9(c). The methods show improvement with lagged $\overline{\text{MIR}}$, providing a broader threshold interval compared to $\overline{\text{MIR}}$. Additionally, lagged-PC permits the correct inference of network structure, unlike PC.

Next, we investigate the influence of parameters on network inference using trajectories from coupled logistic and circle maps. The coupling strength, α, and the time length are two critical parameters that affect the inference of the network. The former impacts the system's dynamics, while the latter ensures sufficient information availability for capturing

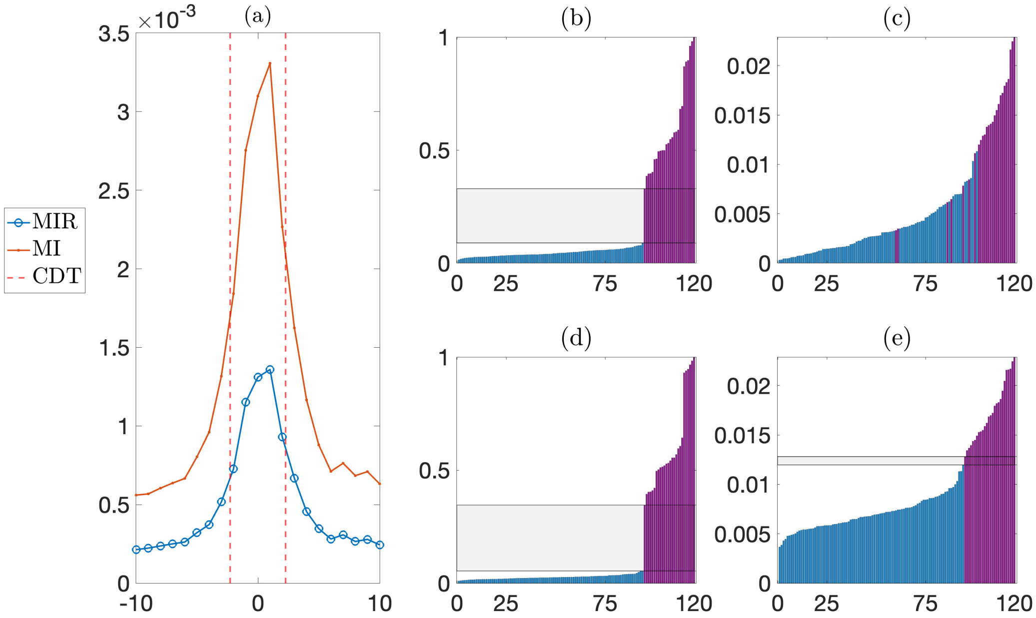

Figure 5.9: **Network inference from a trajectory of coupled circle map**. (a) The values of MI and MIR from a connected pair of nodes are shown as functions of delay, τ. (b) Network inference using $\overline{\text{MIR}}$, (c) using PC, (d) using lagged-$\overline{\text{MIR}}$, and (e) using lagged PC.

connectivity. Hence, in Fig. 5.10, panels (a) and (c) illustrate the effects of these parameters on network inference performance, evaluated using ROC distance (see Eq. (5.19)).

For the data generated from the coupled logistic map with $r = 4$ and 16-nodes network, successful network inference is observed in the dark blue region corresponding to the mid-level of coupling strength and time length greater than 10^4 in Fig. 5.10(a). Stated differently, there is no perfect inference in the interval of coupling strength $[0, 0.04]$ and $[0.12, 0.2]$, possibly due to the following reasons

1. For low coupling strength or short time lengths, the nodes may not exchange sufficient information for $\overline{\text{MIR}}$ to detect links accurately. For example, data generated from the logistic map with a coupling strength of 0.03 and a time length of 10^5 allow perfect network inference (refer to Fig. 5.8), while there is no perfect inference for time lengths shorter than $50,000$ (refer to Fig. 5.10(a)),

2. The dynamics of the system may not be chaotic for some coupling strength, as shown

> in Fig. 5.10(c), where the dynamics of logistic trajectories for the coupling strength in $[0.17, 0.2]$ are either regular or weakly chaotic. If it is a regular trajectory, the data points do not cover the probabilistic space Ω, resulting in elevated MIR values for all pairs. If the trajectory is weakly chaotic, the spreading of data points across the Ω space may take longer (implying high CDT) so, it might require even longer time series length to exchange sufficient information.

Figure 5.10(c) shows that the network inference from the trajectories of the circle map has a wider blue dark area, indicating the perfect inference, compared to the trajectories of the logistic map. For low coupling strengths in $[0, 0.01]$, $\overline{\text{MIR}}$ may not fully capture the relationships between pairs, potentially necessitating a longer time-series length. Figure 5.10(d) shows that the trajectories of the circle map consistently exhibit chaotic behaviour across all coupling strengths considered, α.

5.3.2.1 Additive Gaussian Noise

Real-world data often involve noise, and noise reduction methods cause information loss in network inference using information-based approaches. Therefore, the method should be robust to noise to be able to infer the network from real data. We tested $\overline{\text{MIR}}$ under standard Gaussian noise with mean 0 and variance 1. The noisy data,

$$x_s = f_n(x_n, r, \alpha) + \gamma\xi,$$

where $f(x_n, r, \alpha)$ is a map equation with initial condition, x_n, model parameter r and the coupling strength α. We use coupled logistic and circle maps defined in Eqs. (3.3), (3.6), (3.12). Here, $\gamma \in [0, 1]$ is the noise strength and ξ is a random number drawn from the standard normal distribution, i.e., $\xi \sim \mathcal{N}(0, 1)$.

Figures 5.11(a) and (b) show the performance of $\overline{\text{MIR}}$ (based on the ROC distance in Eq. (5.19)) on the data generated from the coupled logistic and coupled circle map, respectively, with a time length, 5×10^4. The plots depict the noise strength, γ, on the x-axis, and the coupling strength, α, on the y-axis. In both cases, the perfect network

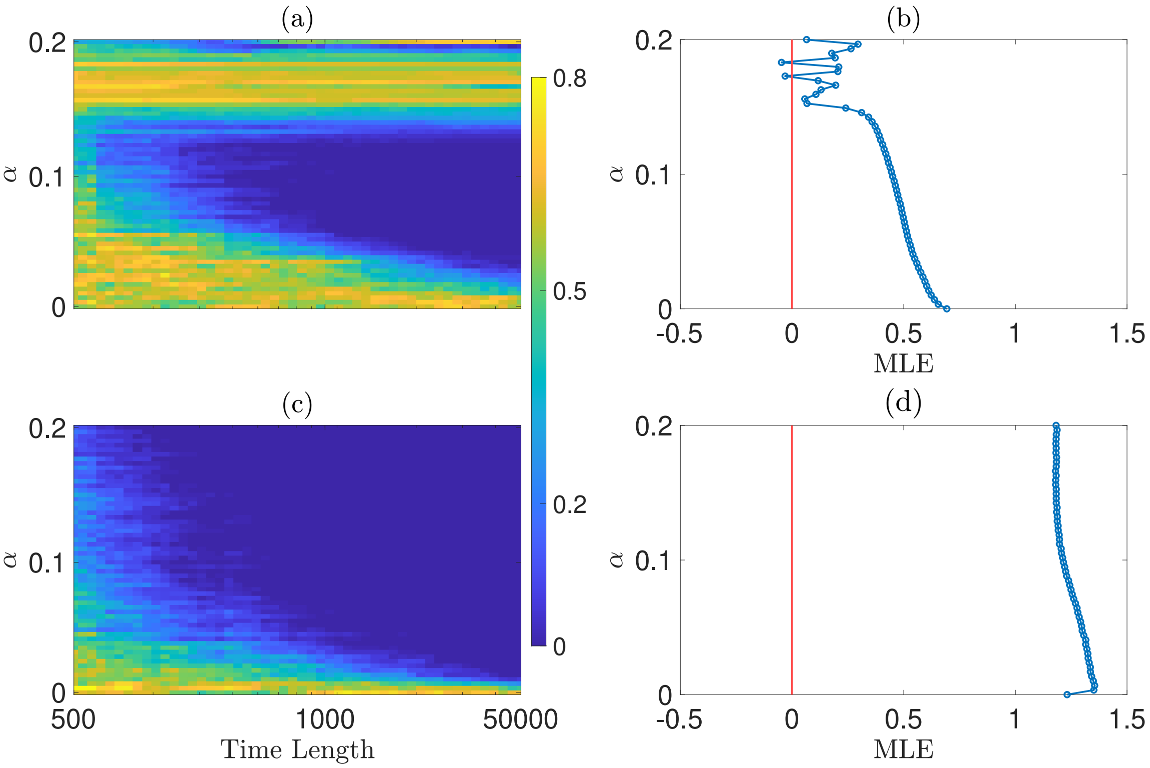

Figure 5.10: **Impact of coupling strength and time length on network inference**. (a) Inference of the network from the trajectories of the logistic map using $\overline{\text{MIR}}$; coupling strengths on the y axis and time lengths on the x axis on the log scale. The colour map indicates the ROC distance: dark blue colour corresponding to the perfect inference as the ROC distance is zero. (b) The variation in MLE of the trajectories of logistic map with respect to the coupling strength, α. (c) Network inference from trajectories of circle map using $\overline{\text{MIR}}$ across coupling strengths on the y-axis and time lengths on the x-axis in logarithmic scale. The colour map indicates the ROC distance: dark blue colour corresponding to the perfect inference as the ROC distance is zero. (d) The variation in MLE of the trajectories of circle map with respect to the coupling strength, α.

inference becomes unattainable when $\gamma > 0.5$, and the ranges of coupling strength leading to perfect inference are the same as those without noise. Hence, we conclude that $\overline{\text{MIR}}$ can accurately capture links under Gaussian noise with $\gamma < 0.5$, provided that the coupling strength allows perfect network inference from noiseless data. However, as the noise becomes stronger, the performance of the method deteriorates, ultimately resulting in no perfect inference for all considered coupling strengths when $\gamma > 0.5$ as there is no cell in the panel (b) denoted by dark blue colour after this point (even though it looks somewhat dark blue for $\gamma > 0.5$, those are not for perfect inference but are very close to

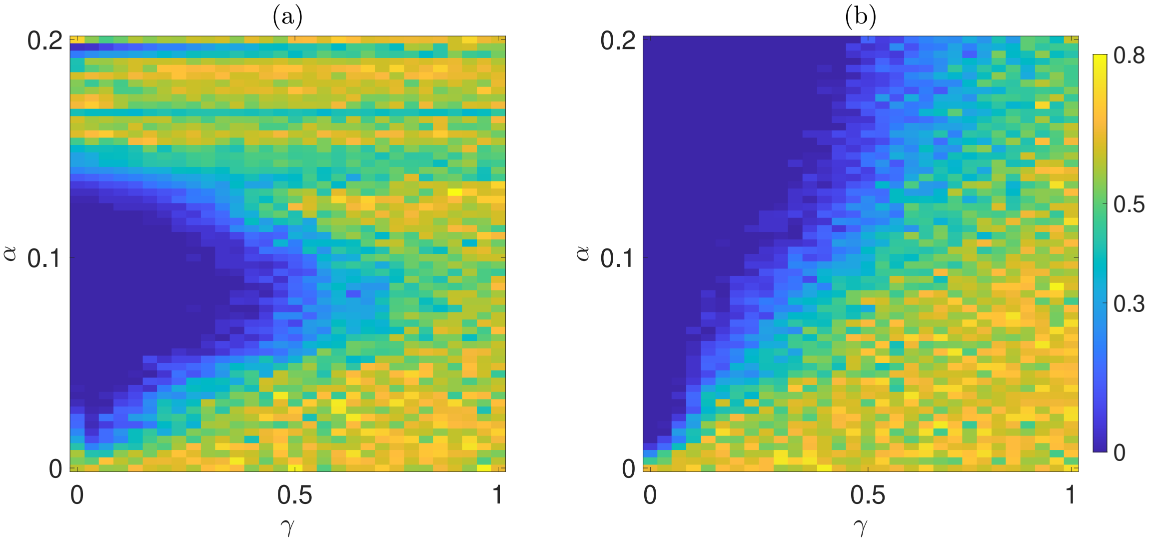

Figure 5.11: **Impact of the Gaussian noise on network inference** The plots depicts the performance of network inference based on the ROC distance in a range of coupling strength, $\alpha \in [0, 0.2]$, and noise strength, $\gamma \in [0, 1]$ from the trajectories of (a) the coupled logistic map and (b) the coupled circle map.

it; we double-checked it with the computed numbers).

5.3.3 Deterministic Continuous Systems

5.3.3.1 The Hindmarsh-Rose System

We generate data from the coupled HR model in Eq. (3.14) with $g_l = 0.1$ using a network of 12-node as shown in Fig. 3.10, with an integration time, $t_f = 2 \times 10^4$, and a step size $h = 0.1$. Among the variables of the HR model, namely, p, q, and n, we selected p as the variable of interest. We applied the $\overline{\text{MIR}}$, PC, lagged-$\overline{\text{MIR}}$, and lagged-PC methods on the p variable of the trajectory, as demonstrated in Fig. 5.12. Taking into account the decrease in mutual information over time-lags after CDT, indicated by the vertical red dashed line, we computed lagged-$\overline{\text{MIR}}$ and lagged-PC across the lag range from -16 to 16.

All methods exhibit similar performance by inferring the network structure within a comparable range of thresholds. The performance of lagged-$\overline{\text{MIR}}$ and lagged-PC, shown in Figs. 5.12(d) and (e), does not show improvement compared to $\overline{\text{MIR}}$ and PC, as illustrated in Figs. 5.12(b) and (c). This lack of improvement is attributed to MI reaching its peak

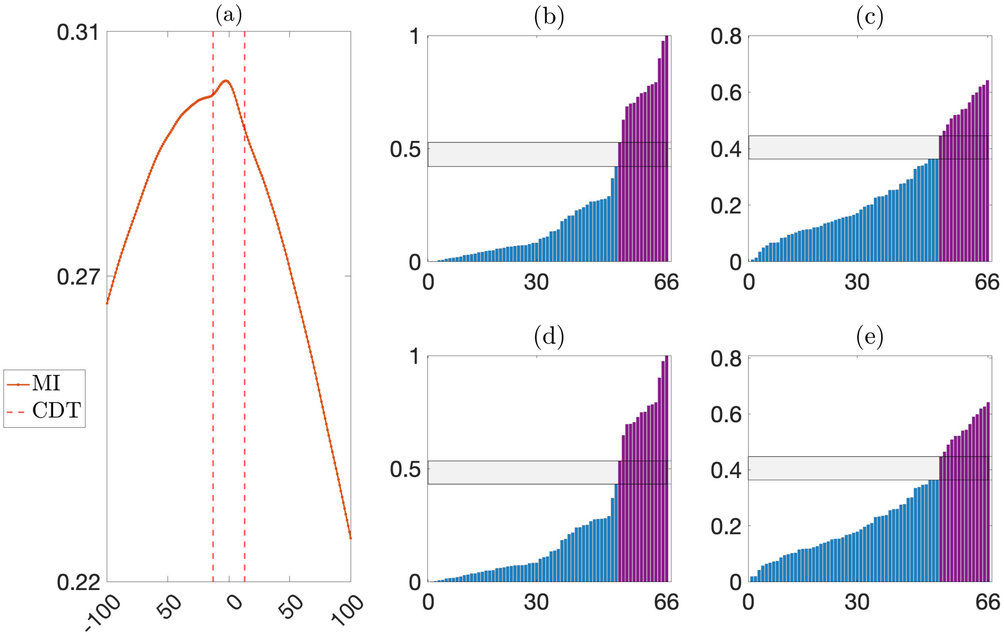

Figure 5.12: **Network inference from a trajectory of coupled HR system** (a) MI values from a connected pair of nodes are shown as functions of lag, τ. (b) Network inference using $\overline{\text{MIR}}$, (c) using PC, (d) using lagged-$\overline{\text{MIR}}$, and (e) using lagged PC.

at a time delay of 0, as seen in Fig. 5.12(a).

The coupled HR model consists of 3 variables, p, q, and n for each node in a network. Although we have previously used p as a probe for network inference, it is also possible to use other variables or combinations thereof to enhance the method's performance in distinguishing connected pairs from unconnected ones. Therefore, we consider the variables q and n, as well as the time derivative of p and the sine of p, as alternative probes, as shown in Fig. 5.13(a)-(d), respectively. All of these probes enable successful network inference, although probes n and $\sin(p)$ exhibit slightly better performance.

5.3.3.2 The Lorenz System

We generated data from the coupled Lorenz system in Eq. (3.16) with coupling strength, $K = 0.2$, adjacency matrix A, of the network of 16 nodes in Fig. 3.10(b). The system is numerically solved using the step size $h = 0.1$ and a final integration time $t_f = 2 \times 10^4$. The trajectory is recorded once the MLE stabilises, based on the assumption that the standard

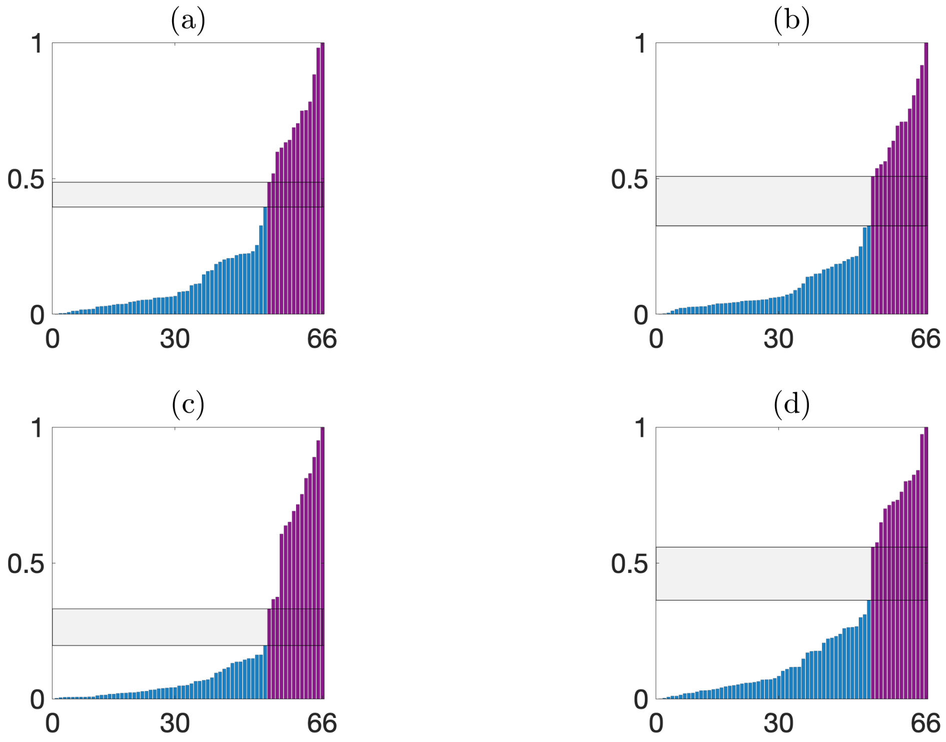

Figure 5.13: **Network inference using MIR from different probes of the coupled HR system** Network inference using MIR from the probes of (a) q variable (b) n variable (c) time derivative of variable p (d) sine of p variable.

deviation of 500 MLEs within the sliding time windows becomes less than or equal to 10^{-2}. Thus, the first 2094 data points were considered transients and discarded.

We implemented the methods of $\overline{\text{MIR}}$, PC, lagged-$\overline{\text{MIR}}$, and lagged-PC on the x variable, can be seen in Fig. 5.14(b)-(e), respectively. Taking into account the decrease in MI over time-lags after CDT (see Fig. 5.14(a)), indicated by the vertical red dashed line, we computed lagged-$\overline{\text{MIR}}$ and lagged-PC over range of time lags from -8 to 8. All methods exhibit similar performance by inferring the network structure correctly within a comparable range of thresholds. The performance of lagged-$\overline{\text{MIR}}$ and lagged-PC, demonstrated in Fig. 5.14(d) and (e), does not show improvement compared to $\overline{\text{MIR}}$ and PC, illustrated in Fig. 5.12.(b) and (c). This lack of improvement is attributed to MI reaching its peak with a time delay of 0 (see Fig. 5.14(a)). We also used the different probes in network inference: variable y, and z, the time derivative of variable x and sine of variable x, as presented in

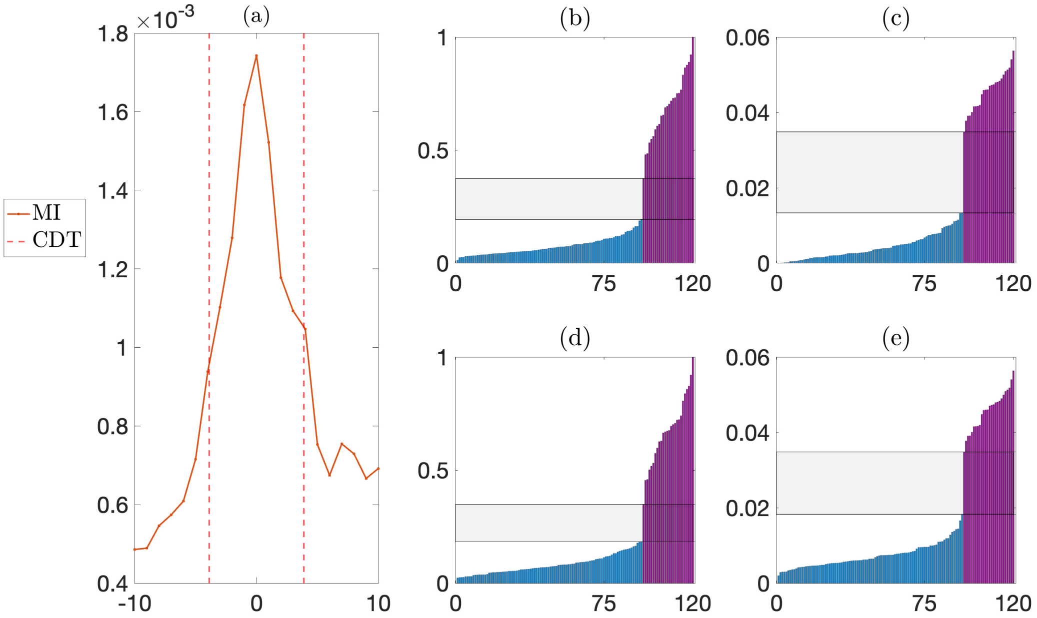

Figure 5.14: **Network inference from a trajectory of coupled Lorenz system** a) MI values from a connected pair of nodes are shown as a function of lag, τ. (b) Network inference using $\overline{\text{MIR}}$, (c) using PC, (d) using lagged-$\overline{\text{MIR}}$, and (e) using lagged PC.

Fig. 5.15(a)-(d), respectively. However, all probes were unable to reconstruct the network structure, as the $\overline{\text{MIR}}$ values of the connected and unconnected ones are mixed. Some probes might improve the performance of network inference. The choice of the best probe depends on its ability to distinguish connected and unconnected pairs.

5.4 Dynamics on Network Models

We generate the data from the dynamics of coupled logistic and circle maps, coupled Lorenz system, and coupled Kuramoto phase oscillators based on the network structures generated from the network models of Erdös-Rényi random, Watts-Strogatz small world and Barabási-Albert scale free networks as discussed in the previous section. We presented MLE of the system, Kuramoto phase order, ρ, and ROC distances (see Eq. (5.19)) of the network inferences using MIR across the coupling strengths. We investigate the question of how the types of network affect the dynamics of the system and network inference in

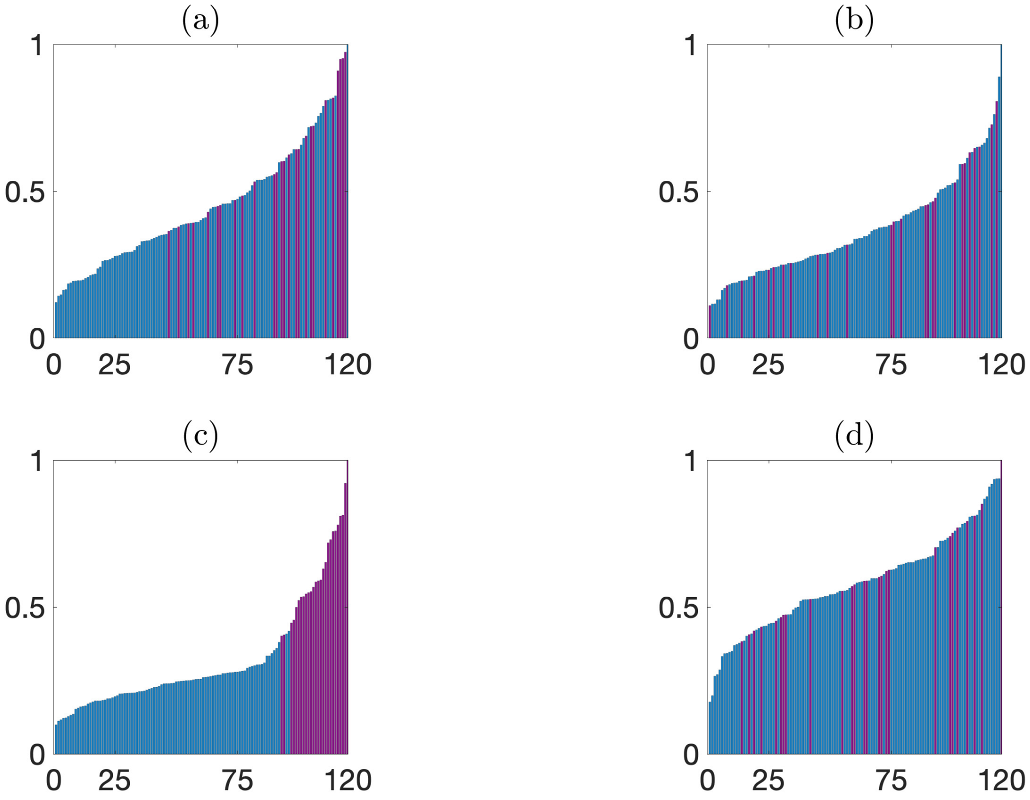

Figure 5.15: **Network inference using $\overline{\mathrm{MIR}}$ from different probes of the coupled Lorenz system** Network inference using $\overline{\mathrm{MIR}}$ from the probes of (a) y variable (b) z variable (c) time derivative of variable x (d) sine of x variable.

relatively small networks of 40 nodes.

5.4.1 Coupled Logistic Map

We generate the data from the coupled logistic map using Eqs. (3.6) and (3.13) based on the ER network in Fig. 2.4(a), SW in Fig. 2.5(a), and SF in Fig. 2.6(a). The 5×10^5 data points recorded after the transient period in which the standard deviation of 500 MLEs within a time sliding window becomes less than a predefined threshold value 10^{-3}. We consider the coupling strengths at 21 equidistant points in the interval $[0, 1]$. The dynamics from the ER network shows a regular trajectory at three coupling strengths where MLE is less than zero, indicated by the horizontal red line, in Fig. 5.16(a). However, trajectories from the network types, SW and SF, the dynamics of the systems are chaotic

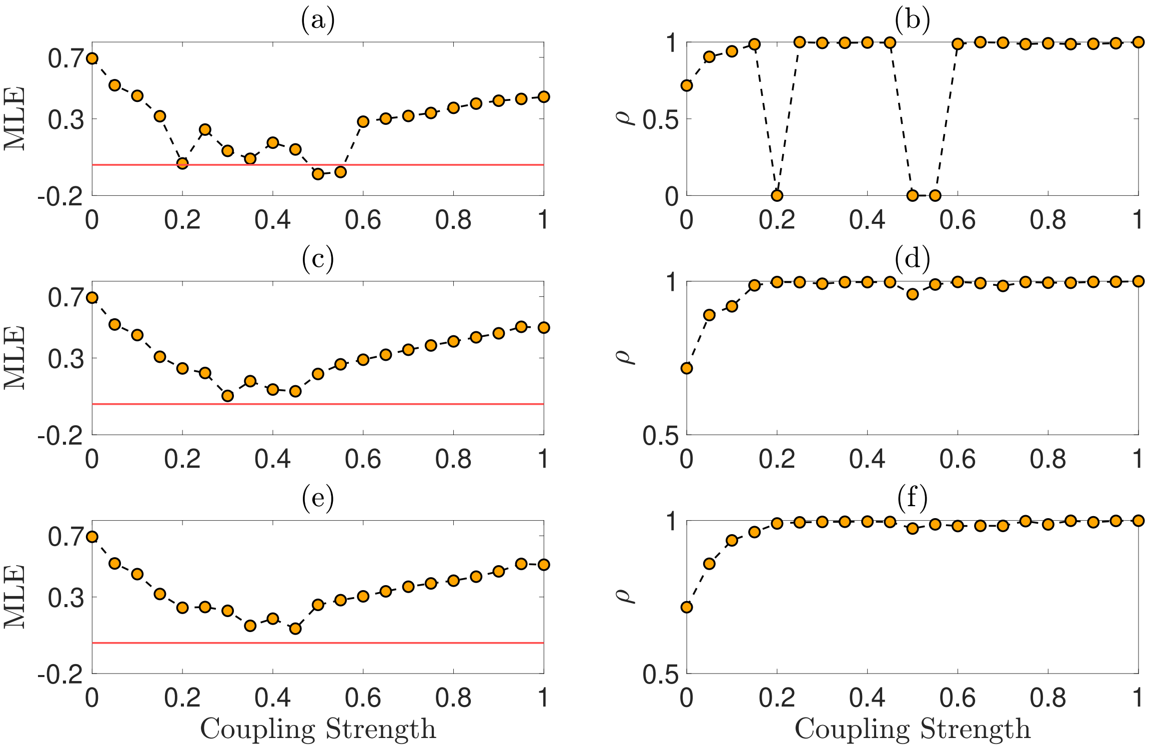

Figure 5.16: **Dynamics of coupled logistic map on network types**. Panels (a), (c), (e) show MLE of the system across coupling strength for the data generated from ER, SW and SF network types, respectively. Panels (b), (d), (f) show that the Kuramoto phase order ρ in the same order.

for all coupling strengths considered.

The Kuramoto phase order, ρ, of the regular trajectories is zero, whereas it is high for chaotic trajectories. However, for coupling strengths, 0, 0.1, and 0.15, the ρ values are slightly lower at 0.72, 0.90, and 0.94, respectively (see Fig. 5.16(b)). Trajectories generated from network types such as SW and SF show, in general, a high level of phase synchronisation with their Kuramoto phase order, ρ values follow a similar pattern for all coupling strengths of interest, as seen in Figs. 5.16(c) and (e).

As the main assumption in the presented theory of MIR is that the X, Y variables are random, we need to ensure the deterministic dynamics is chaotic. This is because such a system loses its memory after some time and behaves as if it is stochastic. Therefore, in the network inference, we dropped the regular trajectories, which is the reason for disconnection in the blue dashed line in Fig. 5.17. We consider three probes in the network inference: (1) the trajectory itself and (2) its instantaneous phases using the concept of analytic signal

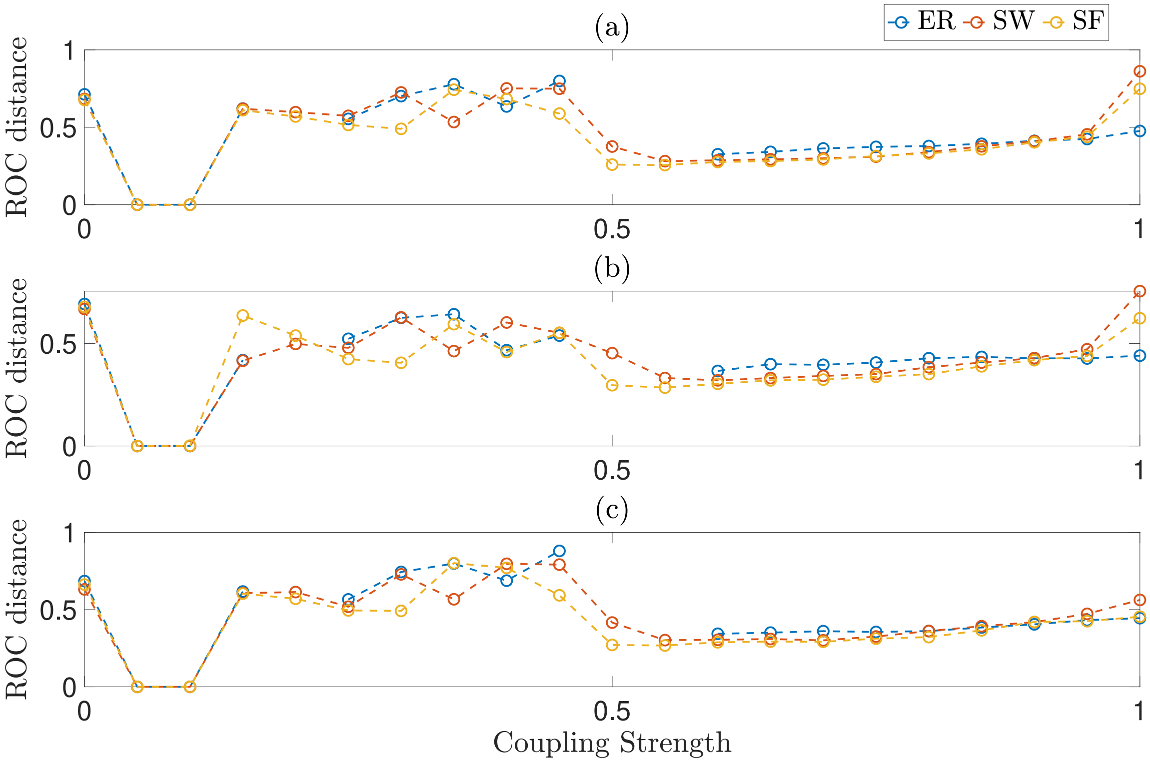

Figure 5.17: **Network inference from the coupled logistic map** (a) ROC distance of the network inference using the signal itself as a probe. Results for ER, SW and SF have been shown with blue, red, and yellow dashed-lines, respectively. (b) using instantaneous phases as probe. (c) using instantaneous frequencies as a probe.

in Sec. 4.3.3 (3) its instantaneous frequencies derived from the instantaneous phases using Eq.(4.5), as shown separately in panels (a), (b), (c) in Fig. 5.17, respectively. We show different types of networks with different colours in all panels, blue for ER, red for SW and yellow for SF. In all network types and probes, we see that MIR can achieve successful network inference in the coupling strengths 0.05 and 0.1. For all other coupling strengths, the trajectories of the three types of network and probes show similar performance in network inference.

5.4.2 Coupled Circle Map

We generate the data from the coupled circle map (see Eqs. (3.12) and (3.13)) based on the networks of ER in Fig. 2.4(a), SW in Fig. 2.5(a), and SF in Fig. 2.6(a). The 5×10^5 data points recorded after the transient period, at the time point at which the standard deviation

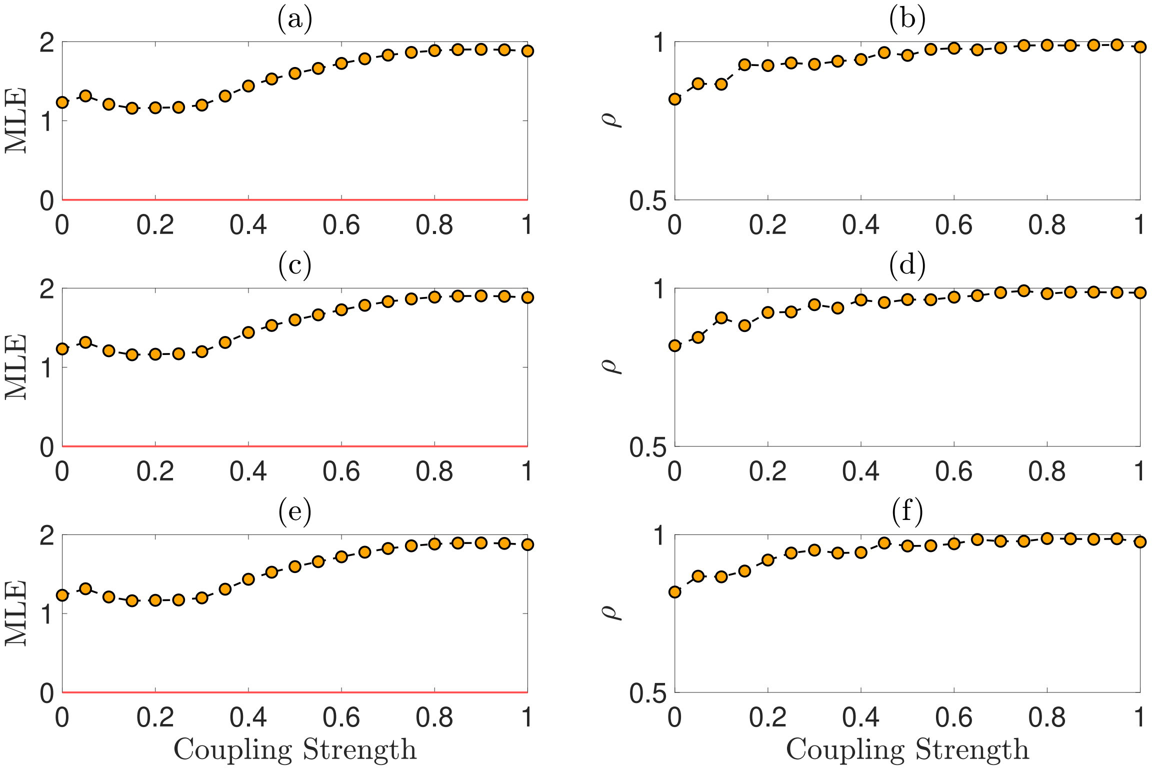

Figure 5.18: **Dynamics of coupled circle map on network types**. Panels (a), (c), (e) show MLE of the system across coupling strength for the data generated from the ER, SW and SF network types, respectively. Panels (b), (d), (f) show that the Kuramoto phase order, ρ, in the same order.

of 500 MLEs within a sliding time window becomes less than the predefined threshold value, 10^{-3}. We consider the coupling strengths at 21 equidistant points in the interval $[0, 1]$. All trajectories for the interest of coupling strengths show similar chaotic dynamics and their Kuramoto phase order, ρ, in the general sense, increases as the coupling strength increases, and it takes values in the interval $[0.82, 0.99]$ indicating highly phase-synchronised data.

In network inference, we consider three probes: (1) the trajectory itself, (2) the instantaneous phases, and (3) the instantaneous frequencies, as shown separately in the panels (a)-(c) in Fig. 5.19, respectively. In general sense, we can see that the ROC distances from different types of network and probes follow a similar pattern. When we consider the trajectory itself as a probe (see Fig. 5.19(a)), MIR achieves perfect inference in all network types for the coupling strengths in the interval $[0.05, 0.65]$. Small-world also achieves a perfect inference for the coupling strength 0.70, whereas the other two do not. If we use

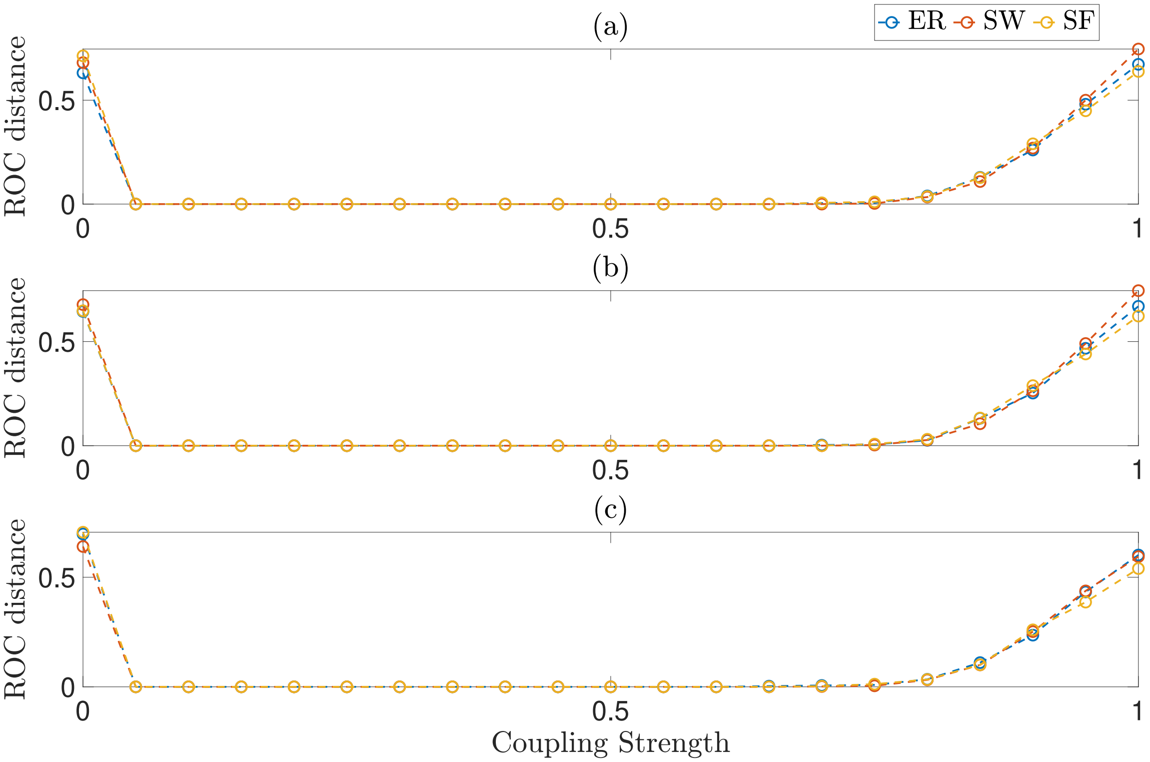

Figure 5.19: **Network inference from the coupled circle map** (a) ROC distance of the network inference using the signal itself as a probe. Results for ER, SW and SF have been shown with blue, red, and yellow dashed-lines, respectively. (b) using instantaneous phases as probe. (c) using instantaneous frequencies as probe.

the instantaneous phases as probe (see Fig. 5.19 (b)), while all of the networks provide perfect inference for the coupling strength in the interval $[0.05, 0.65]$, SW and SF also infer the network correctly for the coupling strength 0.70. If we use instantaneous frequencies as a probe, all types of networks achieve perfect inference in the interval $[0.05, 0.60]$, additionally, network inference from the SW and SF is successful in the coupling strength 0.65. Finally, we can conclude that the different types of networks produce similar dynamics, implying a similar performance of network inference in all probes even if the SW and SF as network types and the instantaneous phases as a probe perform slightly better.

5.4.3 Coupled Lorenz System

We record 5×10^5 data points as a numerical solution of the system in Eq. (3.16) with the step size 0.1, after the transient period, that the MLE stabilised based on the standard

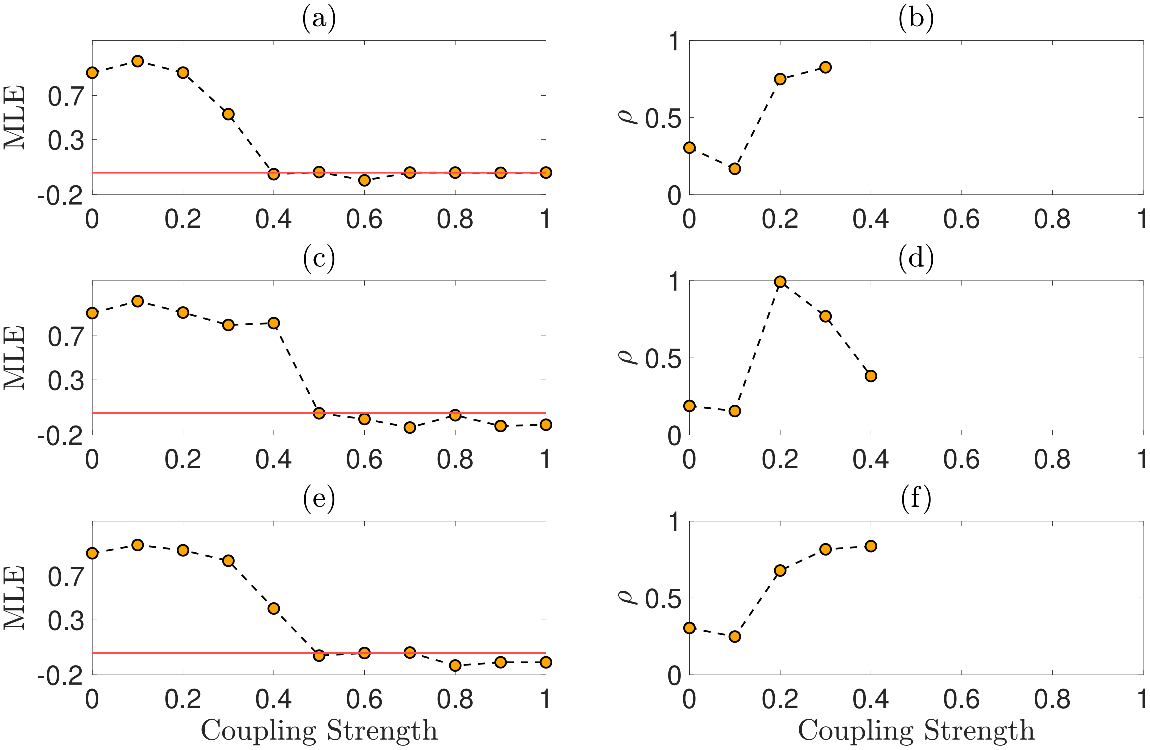

Figure 5.20: **Dynamics of coupled Lorenz system on network types**. Panels (a), (c), (e) show MLE of the system across coupling strength for the data generated from the ER, SW and SF network types, respectively. Panels (b), (d), (f) show that the Kuramoto phase order, ρ, in the same order.

deviation of 500 MLEs within the sliding time window, becoming less than the predefined threshold, 10^{-3}. We used the variable x, the instantaneous phases calculated from the analytic signal of the variable x by HT and the instantaneous frequencies computed from the instantaneous phases using Eq. (4.5). We dropped the regular trajectories in the computation of the Kuramoto phase order ρ and the network inference for the same reason.

Using different network types in the data generation from the Lorenz system results in different dynamics (see Fig. 5.20). The system has positive MLE for the coupling strengths in $[0, 0.3]$ in ER, whereas, in SW and SF, it has positive MLE values for the coupling strength in $[0, 0.4]$ (see panels (a), (c), (d) in Fig. 5.21). Kuramoto phase order, ρ, peaks in the SW for the coupling strength 0.2 as seen in Fig. 5.20 (d). Erdös-Renyi and SF have similar values ρ (see Fig. 5.20 (b), (f)).

Panels (a), (b), and (c) in Fig. 5.21 show the performance of the network inference from the variable x, its instantaneous phases calculated from the analytic signal using

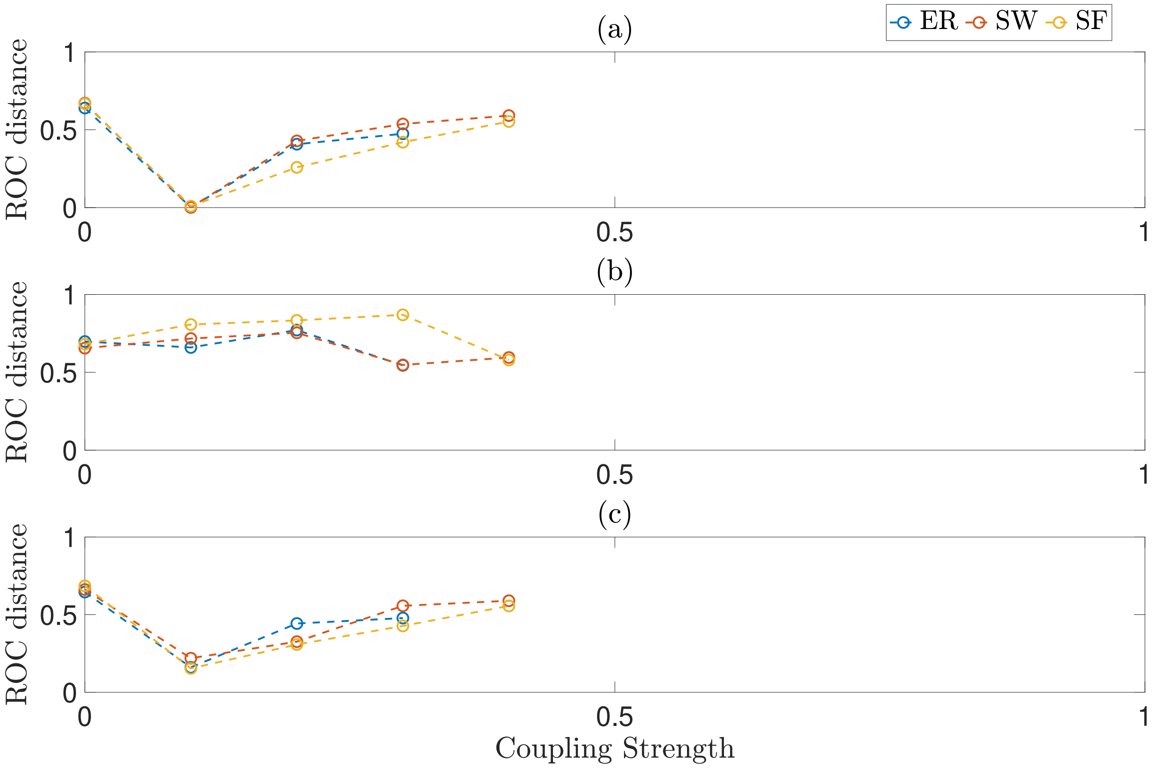

Figure 5.21: **Network inference from the coupled Lorenz system** (a) ROC distance of the network inference using the signal itself as a probe. Results for ER, SW and SF have been shown with blue, red, and yellow dashed-lines, respectively. (b) using instantaneous phases as probe. (c) using instantaneous frequencies as probe.

HT, and its instantaneous frequencies computed from the instantaneous phases in the Eq. (4.5), respectively. Using trajectories as probes in all types of networks performs best for the coupling strength 0.1 with the pair of (TPR, FPR) = $(1, 0.0016)$ (see Fig. 5.21)(a). In overall, SF performs slightly better than the ER and SW. Fig. 5.21(b) shows that the instantaneous phases perform poorly in all types of networks. Although the instantaneous frequencies as the probes perform better than the instantaneous phases but not the trajectories themselves (see Fig. 5.21 (c)).

5.4.4 Coupled Kuramoto Oscillators

We record 5×10^5 data points as a numerical solution of the system in Eq. (3.17) with the step size 0.1, after the transient period, that the MLE stabilised based on the standard deviation of 500 MLEs within the sliding time window falling below the predefined

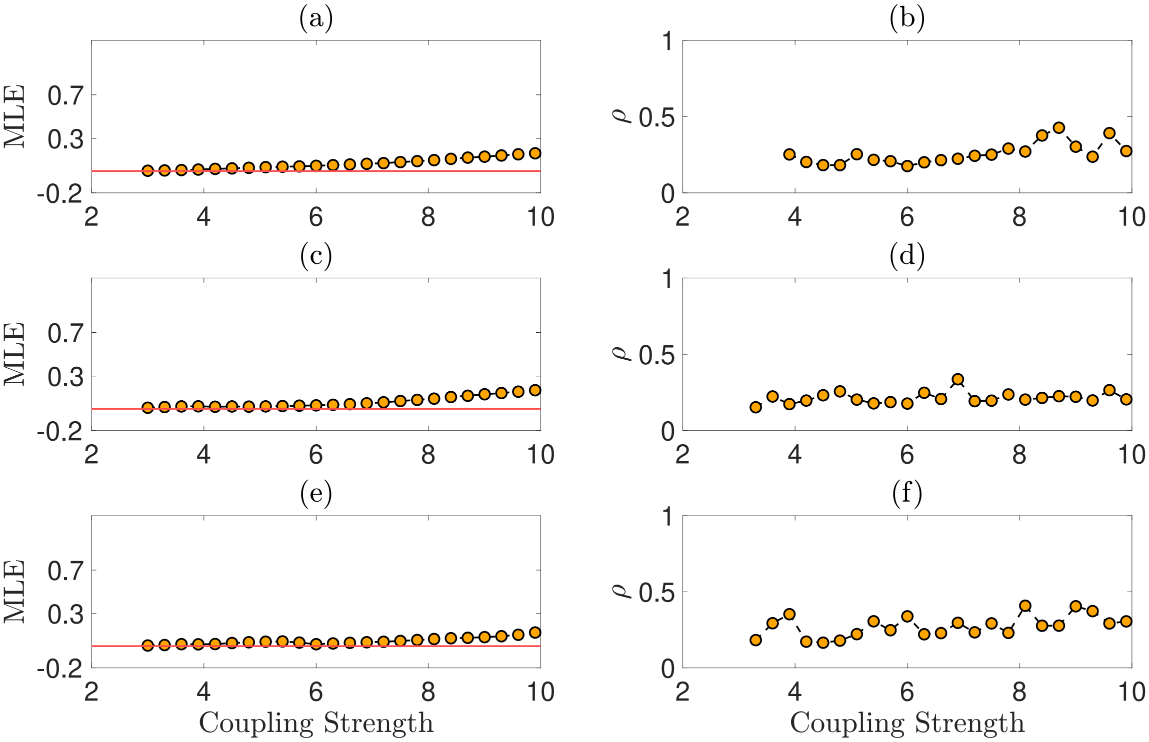

Figure 5.22: **Dynamics of the coupled Kuramoto oscillators on network types**. Panels (a), (c), (e) show MLE of the system across coupling strength for the data generated from the ER, SW and SF network types, respectively. Panels (b), (d), (f) show that the Kuramoto phase order, ρ, in the same order.

threshold, 10^{-3}. From the Kuramoto phases, we compute its instantaneous frequencies from the Eq. (4.5) to use as a probe. There are 24 equidistant coupling strength in $[3, 10]$ considered. To eliminate the weak chaotic behaviour, we drop the trajectories if their MLE is less than 0.01, which is shown with a red line in Fig. 5.22(a), (c), (d). The first three coupling strength are dropped in ER, the first coupling strength is dropped in SW and SF. Fig. 5.22 shows that the values of ρ fluctuate in $[0.15, 0.40]$, indicating low global phase synchronisation.

Fig. 5.23 (a), (b) show the performance of the network inference from Kuramoto phases and their instantaneous frequencies, respectively. It is obvious that the instantaneous frequencies are considerably superior to the phases; however, none can achieve successful inference.

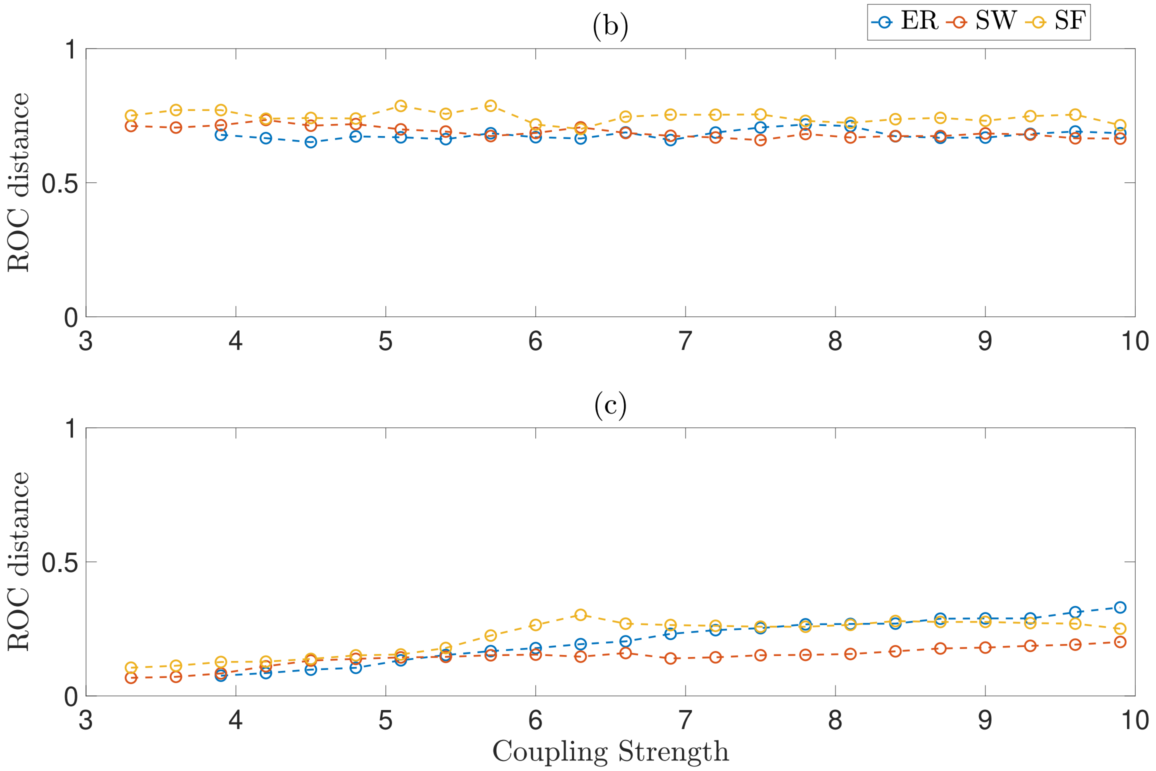

Figure 5.23: **Network inference from the coupled Kuramoto oscillators** (a) ROC distance of the network inference using the Kuramoto phases. Results for ER, SW and SF have been shown with blue, red, and yellow dashed-lines, respectively. (b) the instantaneous frequencies as probe.

5.4.5 Impacts of CS and PS on Network Inference

We presented two cases of synchronisations in a network of 5 nodes in sections 4.2.3 and 4.3.4, that we observed CS, PS and IPS. In the case of a trajectory from the coupled logistic map, we observed CS between nodes 1 and 3 after iteration 6×10^4 (see discussion in Sec. 4.2.3). In the case of a trajectory from the coupled Rössler, PS and IPS occur among some pairs in the network (see discussion in Sec. 4.3.4).

In this section, we investigate the effects of synchronisation on network inference using MIR in these two cases. In particular, to examine the impact of CS, we compare the MIR of whole data in Sec. 4.2.3 with MIR of the same data before CS occurs. Furthermore, to assess the effect of PS and IPS, we compare MIR of phase-synchronised data in Sec. 4.3.4, with MIR of its TWSD, which destroy the phase synchronisation by randomising the instantaneous phases obtained by FT (see detail in Sec. 6.1.2.2).

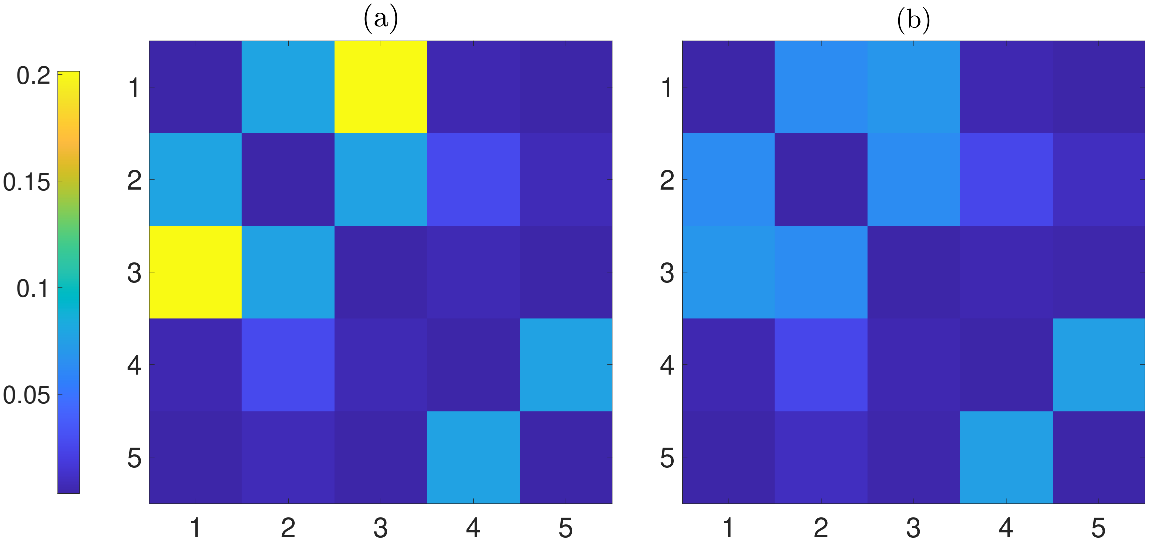

Figure 5.24: **The impact of CS on MIR** (a) MIR values of the pairs in whole data, CS between nodes 1 and 3 results in significantly higher MIR value than others. (b) MIR values before CS occurs. We used first 6×10^5 data points to compute MIR between pairs. MIR between nodes 1 and 3 is now comparable with others.

5.4.5.1 Complete Synchronisation in the Logistic Map

We observed that nodes 1 and 3 behave as one after $60,000$ iterations (see Fig. 4.4(c) and (d)), and their magnitude order is the highest among all pairs, as seen in Fig. 4.11(b). Complete synchronisation between nodes 1 and 3 results in an significantly large MIR value compared to other pairs as shown in Fig. 5.24(a), whereas Fig. 5.24(b) shows that the MIR between the pairs 1, 3 is comparable with MIR of other pairs before CS occurs, when we consider first 6×10^4 data points. To mitigate the effect of CS on network inference, one might consider excluding the fully synchronised time interval. However, in this example, it still does not provide a successful network inference due to high correlation between nodes 1 and 3, even though they are not connected.

5.4.5.2 Phase Synchronisation and Intermittent Phase Synchronisation in the Rössler System

We observed PS and IPS among some pairs in a trajectory of the Rössler system, while the rest remained asynchronous, as shown in Fig. 4.10. Furthermore, we found that the pairwise phase orders of the synchronised pairs are significantly higher than those of the others, as seen in Fig. 4.12(b). From previous observations, we also know that although the phase order is the highest for the synchronised pairs, their PC does not mirror this relation because it does not consider nonlinear relations (see Fig. 4.12(b) and (c)). For example, the phases of nodes 1 and 3 are locked but their PC is relatively lower than one of the asynchronous pairs, 1-4.

Next, we will examine the influence of phase synchronisation on network inference using MIR. Therefore, we compare MIRs among the pairs in original trajectory with those in TWSD, which randomises the phases of a trajectory by the Fourier transform while preserving the linear relations. Lacking the phase relation in the data, we can observe its influence on the MIR. Figure 5.25(a) demonstrates the MIR among the pairs of original data and the synchronous pairs, 1-2, 1-3, 2-3, and 4-5, have the highest MIRs among all implying that the MIR can detect the phase relation. Recall that PC of asynchronous pair, 1-4, was higher than that of the synchronous pair, 1-3. From the comparison of MIRs of the original data and the TWSD in Fig.5.25(b), we can observe that the MIR of all pairs decreases when the phase relation is destroyed. This provides further evidence that phase synchronisation contributes to MIR value.

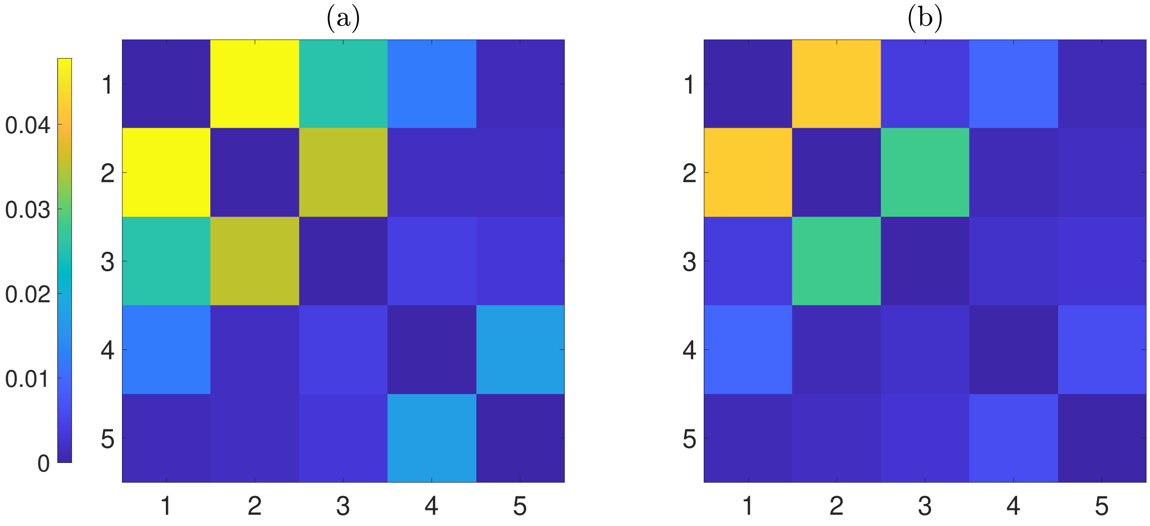

Figure 5.25: **The impact of PS on MIR** (a) MIR values of the phase-synchronised trajectory of Rössler, discussed in Sec. 4.3.4 (b) MIR values of TWSD generated from the trajectory of the Rössler system (4.12) by randomising Fourier phases.

Finally, we can conclude that the phase and the amplitude relation affect MIR values.

The proposed approach, APMSD introduced in Chapter 7, comes from this observation.

5.5 Summary

In this chapter, we reviewed the theoretical background of information-theoretical approaches. We began by explaining the concept of Shannon entropy and defined mutual information and mutual information rate. From a computational perspective, we introduced the binning method to compute probability values from time series data.

To infer network topology from the data, we employed MI, MIR, double-normalised MIR, PC, lagged double-normalised MIR and lagged-PC on Gaussian-distributed correlated data from a given covariance matrix, discrete maps, coupled logistic and circle, continuous systems, coupled HR and Lorenz. Our findings indicated that lagged versions of the methods outperform those without lag, if MIR of a connected pair at lag τ is higher. We also demonstrated the double-normalised MIR is robust to Gaussian noise, enabling the implementation of information-based approaches on real-life data which are commonly noisy. Furthermore, we discussed two factors influencing the performance of network inference, coupling strength, and time length. It is possible to use different probes in network inference, we used the second and third variables of the HR and Lorenz systems, in addition to the time derivative and sine of the first variable following [64]. They successfully inferred the network in the HR but they did not in the Lorenz system. We concluded that the best choice of probe depends on its ability to distinguish connected and unconnected pairs.

We have already introduced and generated some of the most well-known network models, including random graphs, small-world networks, and scale-free networks of 40 nodes in Chapter 2. Using these different network types, we generated data from coupled discrete (logistic and circle maps) and continuous (the Lorenz and Kuramoto systems) time models. Regarding their dynamics, we did not observe significant differences across the network structures; in general, they exhibit similar dynamical behaviour and Kuramoto phase order. In terms of network inference, we employed the signal itself, its instantaneous phases using the Hilbert Transform, and its instantaneous frequencies as probes. However, we did not observe significant differences in network inference across the network types. This lack of distinction may be attributed to the relatively small size of our network, consisting of only 40 nodes. Larger networks often exhibit structural properties, such as small-worldness, more prominently.

Finally, to investigate the impacts of CS and PS on MIR, we applied it to two trajectories discussed in Chapter 4: one is from coupled logistic map providing CS and another is from coupled Rössler system providing PS and IPS. Our final conclusion was that both phase and amplitude relation affect MIR values. This observation forms the basis for the proposed method, APMSD, which will be discussed in Chapter 7.

Chapter 6

Network Inference combining MIR and statistical tests

In the context of network inference, the task of thresholding involves converting continuous similarity measures into binary values, representing 0 for unconnected pairs and 1 for connected pairs. The authors in [26] propose assigning the midpoint of the first abrupt change, specifically a 0.1 change, in ordered MIR as the threshold. Because it is very limited to a few cases only, they propose adding a pair of dummy variables, connected through a directed link, to compare their MIRs with the other pairs. However, the dynamics of the dummy pair of nodes may vary, and it affects the MIR values, so it cannot generalise to a broad range of applications. To address the challenges of thresholding, the author in [2] proposed a method that involves comparing MIRs of the original data with those of the surrogate data for the corresponding pairs.

The use of surrogate data is a well-established approach in nonlinear system analysis to study properties such as phase synchronisation [38]. The author in [2] discusses connectivity sources in terms of amplitude correlation and phase synchronisation. He suggests a method of hypothesis testing by selecting appropriate surrogate data based on the source of connectivity present in the system. The fundamental concept behind this method is to compare the MIR values of pairs in the original data with those of pairs in the surrogate data that lack the source of connectivity. If connectivity emerges from the amplitude correlation, one can use RSD to destroy linear and non-linear relations. If the source of connectivity is phase synchronisation, one can use TWSD to destroy phase synchronisation (or non-linear relation) while preserving linear correlations. However, there are two significant drawbacks to this method. First, it can be challenging to determine which type of surrogate data is better for a given time series because being highly amplitude-correlated or highly phase-synchronised does not mean that the nodes are connected through amplitude correlation or phase synchronisation. Second, RSD and TWSD tend to eliminate all forms of correlation and phase synchronisation. This leads to inferring the extra links (resulting in a high FPR) if the pairs of nodes are indirectly connected (meaning that the pairs connected through other node/s) are highly amplitude correlated or highly phase synchronised. To address these drawbacks, we propose a new method in Chapter 7, called the APMSD method, randomises the amplitude and phase at different levels.

This chapter is dedicated to discussing the methods of statistical tests with RSD and TWSD, since the novel method is built on them, considering the different levels of contributions of amplitude correlation and phase synchronisation to MIR. For comparison reasons, the results of statistical tests using RSD and TWSD, and using the APMSD method applied to the same systems with the same parameters and network structure, are presented in this chapter and Chapter 7.

6.1 Statistical Tests and Surrogate Data

6.1.1 Statistical Tests

Hypothesis testing is a well-established method used in dynamical system analysis to determine nonlinear properties such as phase synchronisation [38]. However, the use of surrogate data to identify connectivity patterns in complex networks is proposed by the author in [2]. In this study, the author used RSD, which disrupts the correlation between pairs in the original data. If the connectivity between pairs arises from correlation, employing hypothesis tests with RSD becomes a suitable method to identify links among connected pairs.

Another approach used in the paper is, TWSD which, destroy the phase relationships among variables in time series data. This is achieved by generating surrogate data with randomised phases using the Fourier transform following [53].

Following [2], to infer the connectivity between nodes X and Y from the data using hypothesis tests and surrogate data, one selects a significance level, α and defines the null hypothesis, H_0, and alternative hypothesis, H_1 by,

$$H_0 : \text{There is no connection between nodes } X \text{ and } Y,$$

$$H_1 : \text{There is a connection between nodes } X \text{ and } Y.$$

In this context, the total number of surrogate data, $\aleph_{SD}$, for hypothesis testing should be $\aleph_{SD} \geq \left[\frac{1}{\alpha}\right]$, where $[\cdot]$ denotes the integer part of a number. In this work, we decided to choose $\aleph_{SD} = \left[\frac{1}{\alpha}\right]$ to reduce computational cost. For example, if we intend to test the hypotheses at $\alpha = 0.1$, we produce 10 surrogate data sets. The p-value associated with the null hypothesis, H_0, is,

$$p_{XY} = \frac{\tilde{\aleph}_{SD}}{\aleph_{SD}},$$

where $\tilde{\aleph}_{SD}$ is the number of surrogate data whose MIR_{XY} is higher than or equal to MIR_{XY} of the original data set. Following [2], p_{XY} is the probability of obtaining a test statistic that is equal to or greater than the observed one, assuming that the null hypothesis is true. For example, if $p_{XY} = 0.02$, it would mean that if the null hypothesis is true, there would be a 2% chance of obtaining the observed test statistic or a more extreme one. If this is less than a predefined significance level, α, then we would reject the null hypothesis and say that nodes X and Y are connected. In general, if $p_{XY} < \alpha$, we can reject H_0 and accept H_1, which means that nodes X and Y are connected.

For a network of M nodes with adjacency matrix, $\mathbf{A}$, we repeat the above process for all $\frac{M(M-1)}{2}$ uniquely defined pairs of nodes X and Y, resulting in $\frac{M(M-1)}{2}$ independent hypothesis tests. When k hypotheses are tested simultaneously with a significance level α, the probability of occurrence of false positives (i.e., rejecting the null hypothesis when in fact it is true) is equal to $1-(1-\alpha)^k$, which can lead to a high error rate [56]. This would lead to accepting many false positive connections. Therefore, to control for multiple-testing errors, we use the False Discovery Rate (FDR) method discussed in [9–11]. This will result in an $M \times M$ inferred adjacency matrix, $\tilde{\mathbf{A}}$, for all X and Y pairs with 0 and 1 (binary matrix), where 0 (1) mean no connection (connection) between X and Y. Since $\mathrm{MIR}_{XY} = \mathrm{MIR}_{YX}$, $\tilde{\mathbf{A}}$ is a symmetric matrix with 0s on the diagonal, because we do not consider self connections. The pseudocode of the process of hypothesis tests and surrogate data is given in the algorithm 5.

Algorithm 5 Algorithm for statistical tests

```
Input data of $M$ variables and significance level $\alpha$.
Generate $N_{SD} = \frac{1}{\alpha}$ chosen surrogate data
Compute MIR of original dataset and get $M \times M$ matrix $\mathrm{MIR}^o$
Initialise adjacency matrix, $\tilde{\mathbf{A}}$.
for <i=1:$M$> do                                  ▷ iteration over pairwise comparison
    for <j=i+1:$M$> do
        Initialise $\widetilde{\aleph}_{SD} = 0$
        for <k=1:$N_{SD}$> do                     ▷ iteration over $N_{SD}$ surrogate data
            Compute MIR of surrogate data and get $M \times M$ matrix $\mathrm{MIR}^{SD_k}$.
            if $\mathrm{MIR}^o_{ij} \leq \mathrm{MIR}^{SD_k}_{ij}$ then
                $\widetilde{\aleph}_{SD} = +1$
            end if
        end for
        $p_{ij} = \frac{\widetilde{\aleph}_{SD}}{N_{SD}}$                 ▷ probability of being not connected
        if $p_{ij} < \alpha$ then
            $\tilde{A}_{ij} = 1$
        end if
    end for
end for
Update $\tilde{\mathbf{A}}$ based on FDR process.
$\tilde{\mathbf{A}} = \tilde{\mathbf{A}} + \tilde{\mathbf{A}}^T$                              ▷ as $\mathrm{MIR}_{XY} = \mathrm{MIR}_{YX}$
Return $\tilde{\mathbf{A}}$.
```

Based on the approach used so far, the adjacency matrix $\tilde{\mathbf{A}}$ can be inferred from the data. We used TPR and FPR (see Eq. (5.18)) to assess the performance of the method compared to the adjacency matrix of the original network, $\mathbf{A}$.

6.1.2 Surrogate Data

One should select the surrogate data that lack the source of connectivity. If the source of connectivity is amplitude correlation, one can choose the RSD to destroy the linear and nonlinear relation, while if the source of connectivity is phase synchronisation, one can choose TWSD to destroy phase synchronisation. In this section, we discuss these two surrogate data sets.

6.1.2.1 Random Surrogate Data

Random surrogate data is drawn from a uniformly random distribution, independently of the original data. This method is based on the fact that two random variables, X, Y theoretically have zero correlation. However, in practice, the correlation is very close to zero, resulting in a cloud-like shape resembling non-correlated pairs, as shown in Fig. 5.6. This implies that the data has significantly small MIR values, indicating limited information transfer from one variable to another due to the independence of the variables. Recall that the MIR of two independent random variables X and Y is theoretically equal to zero, but in practice, it is a very small positive number due to the finite number of data and round-off errors in the computation of MIR (see Eq. (5.6)). In this sense, comparing the MIRs of the original data with those of RSD helps eliminate pairs of nodes whose MIRs might come from the round-off errors only, in the computation process.

6.1.2.2 Twin Surrogate Data

Information-based measures are capable of capture nonlinear relations as well as linear relations (see Fig. 5.2). Phase order is one of the characteristics of nonlinear systems having an effect on MIR as discussed in Sec. 5.4.5, where we showed that one of the source of MIR is the phase order by using TWSD, which preserves the linear properties but destroys all phase synchronisation by randomly selecting the phases through FT. Following the process in [53] and using the algorithm provided in [24], we generate TWSD as follows:

Let $x(t)$ be a time series and $\mathcal{F}$ the finite discrete-time FT operator,

$$X(f) = \mathcal{F}\{x(t)\} = \sum_{n=0}^{N-1} x(t)e^{2\pi ift},$$

As discussed in Sec. 4.3.2, FT of $x(t)$ is complex-valued and can be expressed in a polar form,

$$X(f) = A(f)e^{i\phi(f)},$$

where $A(f)$ and $\phi(f)$ are Fourier amplitudes and Fourier phases of the signal, $x(t)$, in the frequency domain, respectively. To obtain a phase-randomised signal $\tilde{x}(t)$, one can rotate the phases, $\phi(f)$, by $\psi(f)$ chosen uniformly in $[0, 2\pi)$ and TWSD of the signal, $x(t)$,

$$\tilde{x}(t) = \mathcal{F}^{-1}\{A(f)e^{i[\phi(f)+\psi(f)]}\}$$

where $\mathcal{F}^{-1}$ is the inverse FT operator.

In the next section, we will present the results from various dynamics, including Gaussian distributed data from a given covariance matrix, discrete-time systems such as logistic and circle maps, continuous-time systems such as Lorenz and HR systems, and stochastic Kuramoto, conducting hypothesis tests using RSD and TWSD.

6.2 Network Inference using MIR and statistical tests

6.2.1 Gaussian Distributed Correlated Data

In this section, we present the results of the statistical test method using RSD on the data generated from a multivariate Gaussian distribution with a given mean vector and covariance matrix, following the d-dimensional probability distribution function in Eq. (5.20) and data are generated in the same way as in sec. 5.3.1.

We demonstrated that PC between variables 8 and 9 is significantly lower (with a value

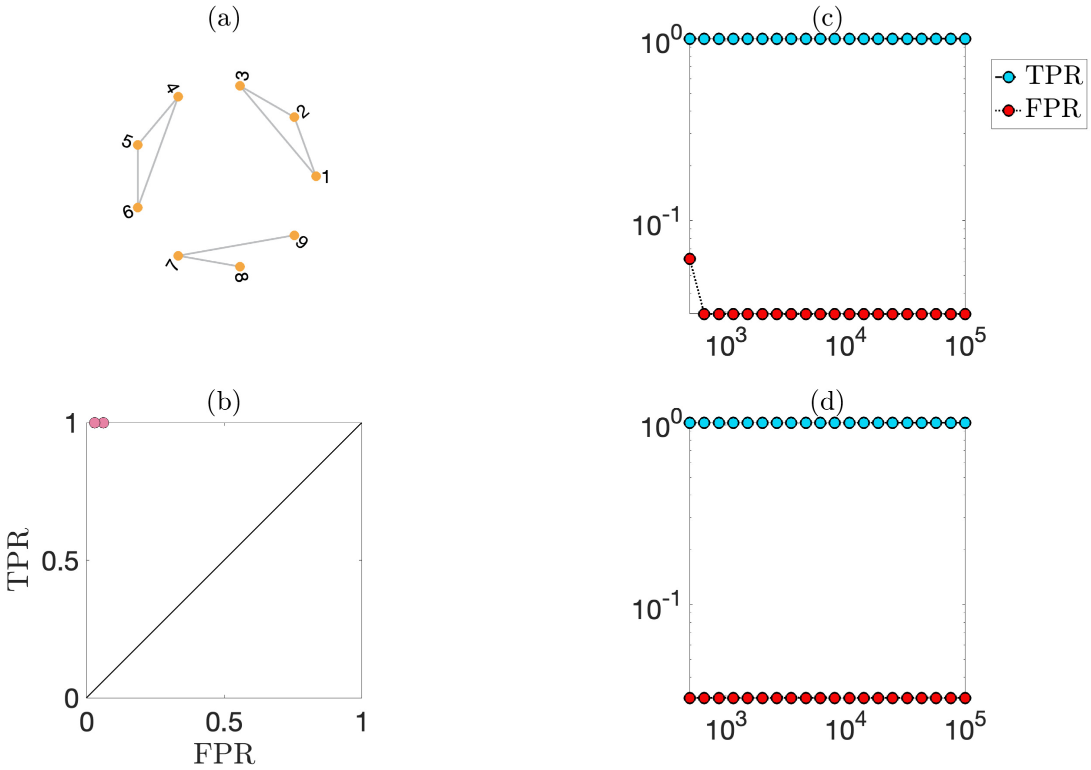

Figure 6.1: **Statistical tests using RSD in Gaussian distributed correlated data.** (a) Network structure between the variables with the assumption of that the pairs connected if corresponding PC is higher than 0.5. (b) ROC plane of TPR and FPR at $\alpha = 0.01$ (c) Evolution of TPR and FPR over time length at $\alpha = 0.1$, represented by blue and red points, respectively. (d) Evolution of TPR and FPR over time length at $\alpha = 0.01$, represented by blue and red points, respectively.

of 0.23) than other pairs and remains significant compared to unconnected pairs. Assuming that there is no link between the pairs whose PC is less than 0.5, we obtain the network presented in Fig. 6.1(a). Hypotheses testing using RSD at 0.1 and 0.01 significance levels are implemented on the data and both cannot achieve successful network inference, as seen in Fig. 6.1 (c) and (d), respectively. Here, the evolution of TPR and FPR over time is shown by the blue and red circles, respectively. Due to the logarithmic scale of the axes in the figure, FPR is not plotted if FPR = 0. However, in our cases FPR always exists, even though TPR reaches to 1, since the method cannot exclude the link between nodes 8 and 9. Therefore, we can conclude that statistical tests method using RSD cannot infer the network structure correctly. Figure 6.1(b) depicts TPR and FPR values on the plane at the significance level, $\alpha = 0.01$, where the point $(1, 0)$ corresponds to perfect inference and the green circle appears if there is perfect inference at any considered time length, however, which is not the case in this example.

6.2.2 Deterministic Discrete Systems

In this section, we present the results of the statistical tests method using RSD on the coupled logistic and circle maps generated from the Eqs. (3.6), (3.12) and (3.13).

6.2.2.1 Logistic Map

We generate the data using Eqs. (3.6) and (3.13) for the coupling strength $\alpha = 0.06$ and the model parameter $r = 4$ and the adjacency matrix, $\mathbf{A}$, of the network of 30 nodes, shown in Fig. 6.2(a). The first 1293 iterations, considered as transients and discarded as the standard deviation of 500 MLEs within a sliding time window becomes less than the predefined threshold 10^{-3} (see discussion in Sec. 3.3.2.1). After transients, 10^5 data points are recorded, and the system's MLE converges to 0.51, which indicates chaotic dynamics.

Figure 6.2 (c), (d) shows the evolution of TPR and FPR over time length, represented by blue and red circles, respectively. We implement hypothesis tests using RSD at $\alpha = 0.1$ and $\alpha = 0.01$. Both cases do not achieve successful network inference, although implementing the method at $\alpha = 0.01$ slightly improves performance. Notably, FPR is not plotted when it is equal to zero, as a consequence of the y axis on logarithmic scale. Figure 6.2(d) shows the plot of the pairs of TPR, FPR at $\alpha = 0.01$ significance level, where no points turn green as an indication of no perfect inference.

6.2.2.2 Circle Map

We iterated the data using Eqs. (3.12) and (3.13) for the parameters $\alpha = 0.03$ and $r = 0.35$ based on the network of 16 nodes, shown in Fig. 6.3(a). After 2060 iterations, 10^5 data points were recorded, as the standard deviation of 500 MLE within a sliding time window becomes less than the predefined threshold, 10^{-3} and MLE converges to 1.33 is an indicator of chaotic behaviour.

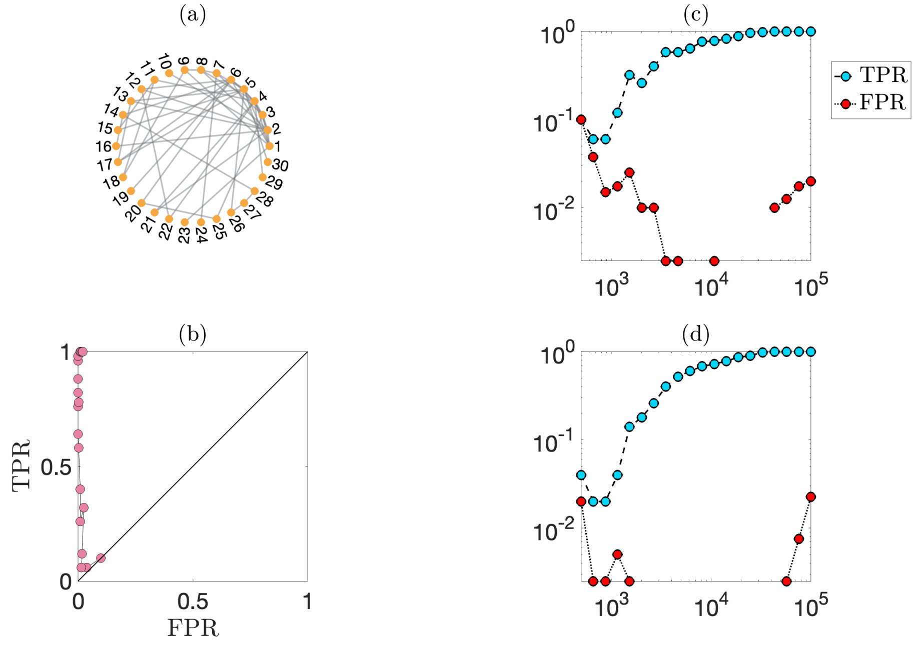

Figure 6.2: **Statistical tests using RSD and coupled logistic maps.** (a) Network structure used in data generation by iterating coupled logistic map. (b) The plot of the pairs of TPR, FPR at $\alpha = 0.01$ significance level. In the case of perfect inference, a point would be located in the upper-left corner and indicated by green circles. (c) Evolution of TPR and FPR over time length at $\alpha = 0.1$, represented by blue and red points, respectively. (d) Evolution of TPR and FPR over time length at $\alpha = 0.01$, represented by blue and red points, respectively. Notably, FPR disappears when it equals zero, a consequence of the logarithmic scaled y-axis. If there is a point corresponds to perfect inference, it is coloured by green.

In Fig. 6.3, panels (c) and (d) show that the hypothesis testing method using RSD captures links correctly as TPR reaches to 1 and FPR 0 at both significance levels $\alpha = 0.1$ and $\alpha = 0.01$. The performance of the method improved at the significance level $\alpha = 0.01$ comparing to at $\alpha = 0.1$ because the FPR lowered for shorter time series. Figure 6.3(b) plots the pairs of TPR, FPR for different time lengths considered with $\alpha = 0.01$, where the pair of TPR, FPR corresponds to the perfect inference indicated by a green circle.

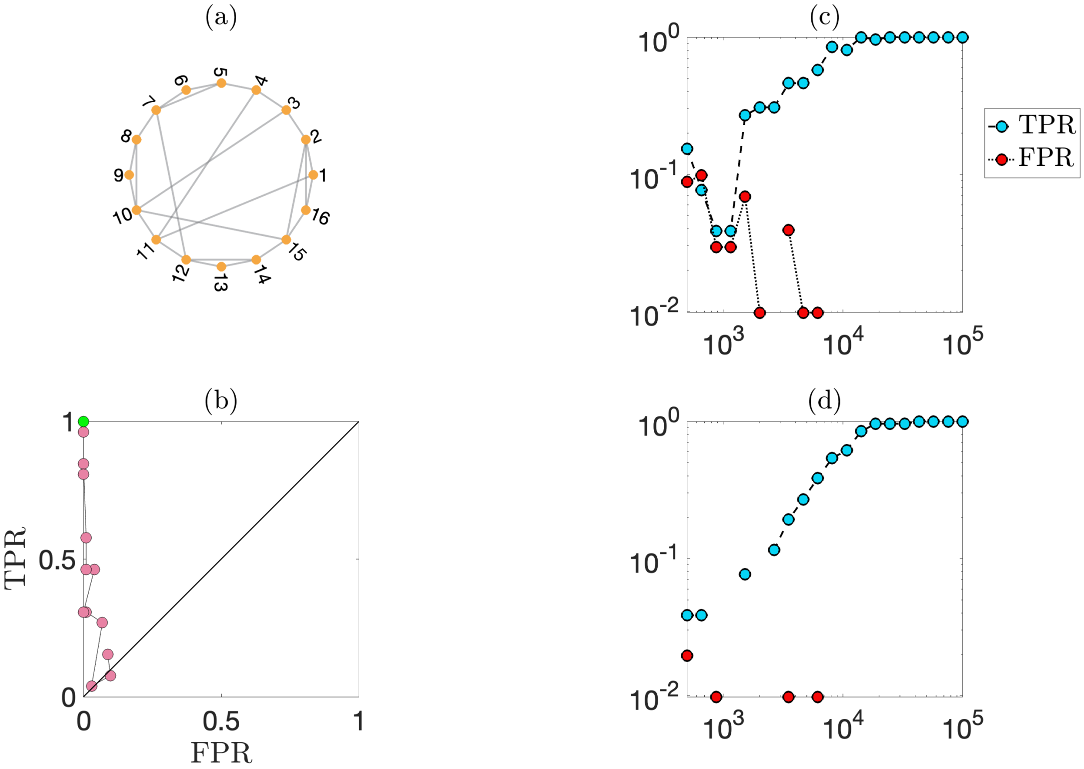

Figure 6.3: **Statistical tests using RSD in circle map.** Panels (a)-(d) present similar plots as in Fig. 6.2.

6.2.3 Deterministic Continuous Systems

6.2.3.1 Lorenz System

The Lorenz system, given by Eq. (3.16), is numerically solved based on the network of 16 nodes, shown in Fig. 6.4(a), with step size, $h = 0.1$, final integration time, $t_f = 2 \times 10^4$, coupling strength, $K = 0.2$. Before recording the time series, the first 2094 data points were considered transient periods since after this point the standard deviation of 500 MLEs within a sliding time window becomes less than the predefined value, 10^{-2}. The system's MLE converges to 0.99, which is evidence of chaotic behaviour. This guarantees that the points will disperse across the probabilistic space, Ω, during the computation of MIR. The variable x is used as a probe in network inference.

We implement the method in the data using both RSD and TWSD as shown in Fig. 6.4 and Fig. 6.5, respectively. In the former, we implement the method using RSD on the

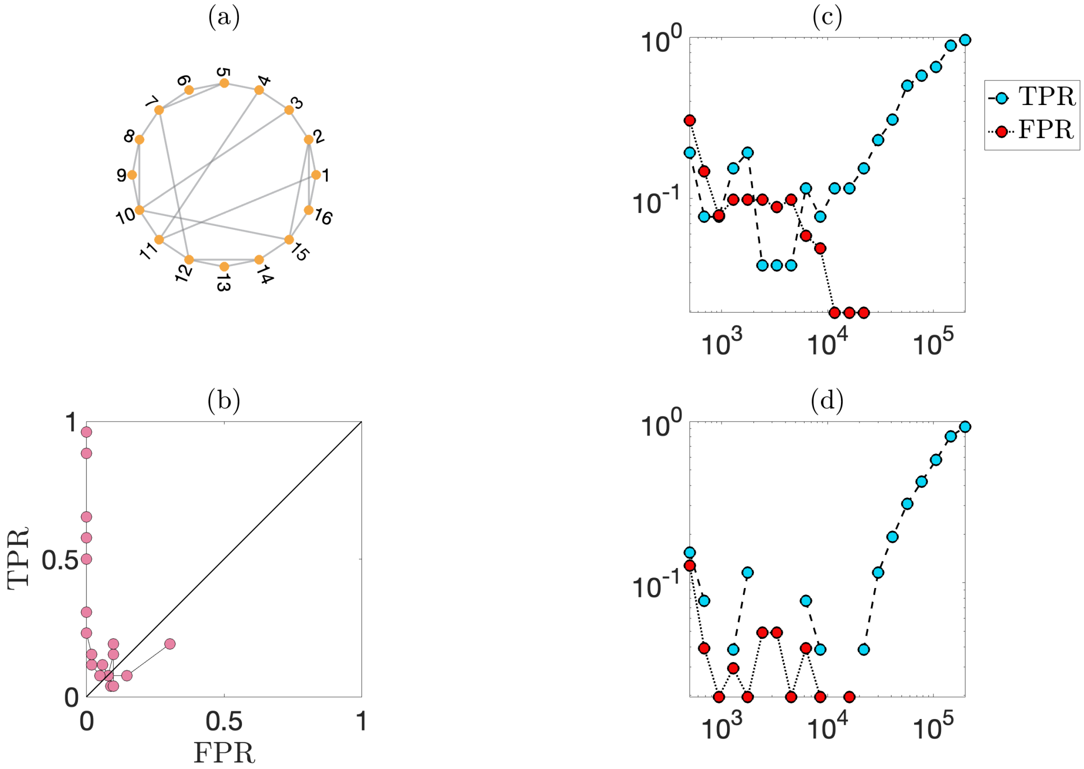

Figure 6.4: **Statistical tests using RSD in Lorenz system.** Panels (a)-(d) present similar plots as in Fig. 6.2.

data at two significance levels, $\alpha = 0.1$ and $\alpha = 0.01$ (see Fig. 6.4(c), (d)). In both, FPR reaches to zero for time length around 3×10^4, however, TPR never reaches 1 even though it continues to improve over the length of the considered time series and eventually reaches its peak at 0.96 for 2×10^5 data points. Therefore, there is no perfect inference in the method of hypothesis tests with RSD at both significance levels $\alpha = 0.1$ and $\alpha = 0.01$. For the shorter time length, we can say that the method at $\alpha = 0.01$ is more conservative in inferring links because both TPR and FPR are smaller, but they eventually produce the same pair of TPR, FPR once the whole data is considered.

In the latter, we implement the method of statistical tests using TWSD at $\alpha = 0.1$ and $\alpha = 0.01$ (see Fig. 6.5(c), (d)); In both, when the whole data is considered, TPR and FPR are significantly high with their values, 0.91 and 0.80, respectively. It shows that the source of connectivity can be explained by amplitude correlation more than phase synchronisation in this system. Since, by removing phase synchronisation, the method results in high FPR

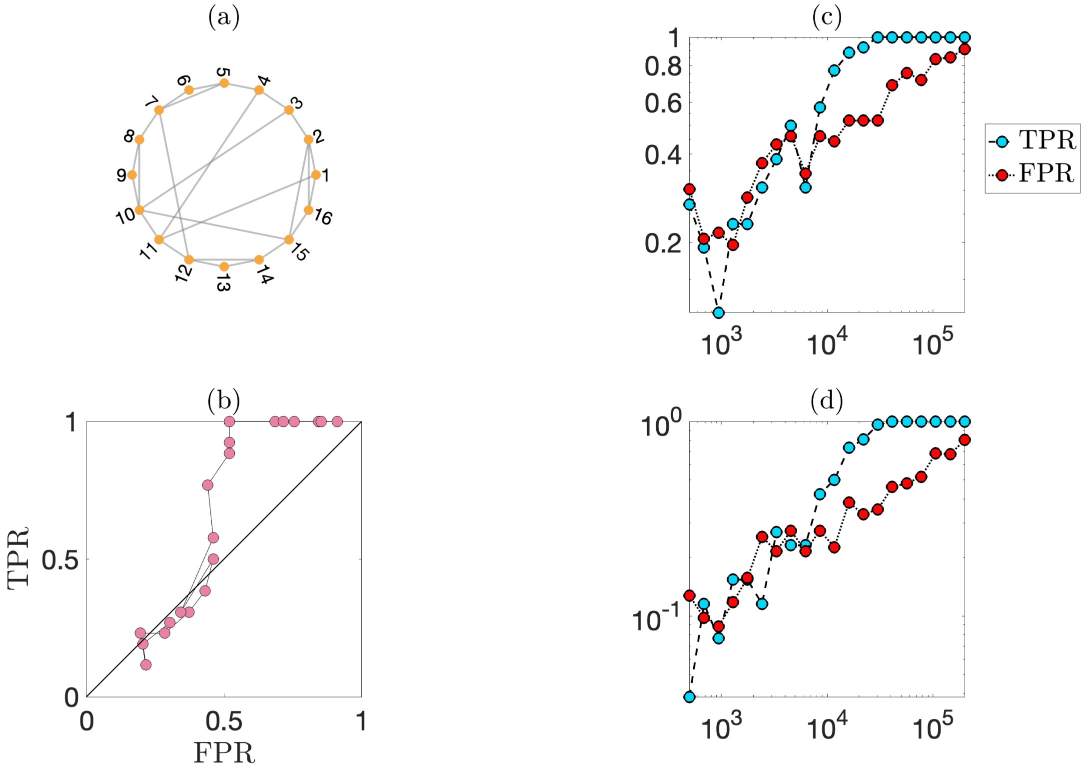

Figure 6.5: **Statistical tests using TWSD in Lorenz system.** Panels (a)-(d) present similar plots as in Fig. 6.2.

implying that unconnected pairs also have a high phase synchronisation. This leads to inferring extra links that are not present in the original network. Figure 6.5(b) plots the pair of TPR, FPR at 0.1 significance level, that is also an evidence of poor performance of the method using TWSD.

6.2.3.2 Hindmarsch-Rose Model

We numerically solved the equation of coupled HR model (see Eq. (3.14)), based on the network of 12 nodes in Fig. 6.6(a), with coupling strength, 0.1. After discarding the first 6641 data points since then, the standard deviation of 500 MLEs within the sliding time window stabilises after this point, falling below a predefined threshold, 10^{-3}. The time series was recorded with the final integration time 2×10^4 and the step size 0.1. Maximum Lyapunov exponent of the system converges to 0.41, which is evidence of chaotic behaviour, and the variable p is used as a probe in network inference.

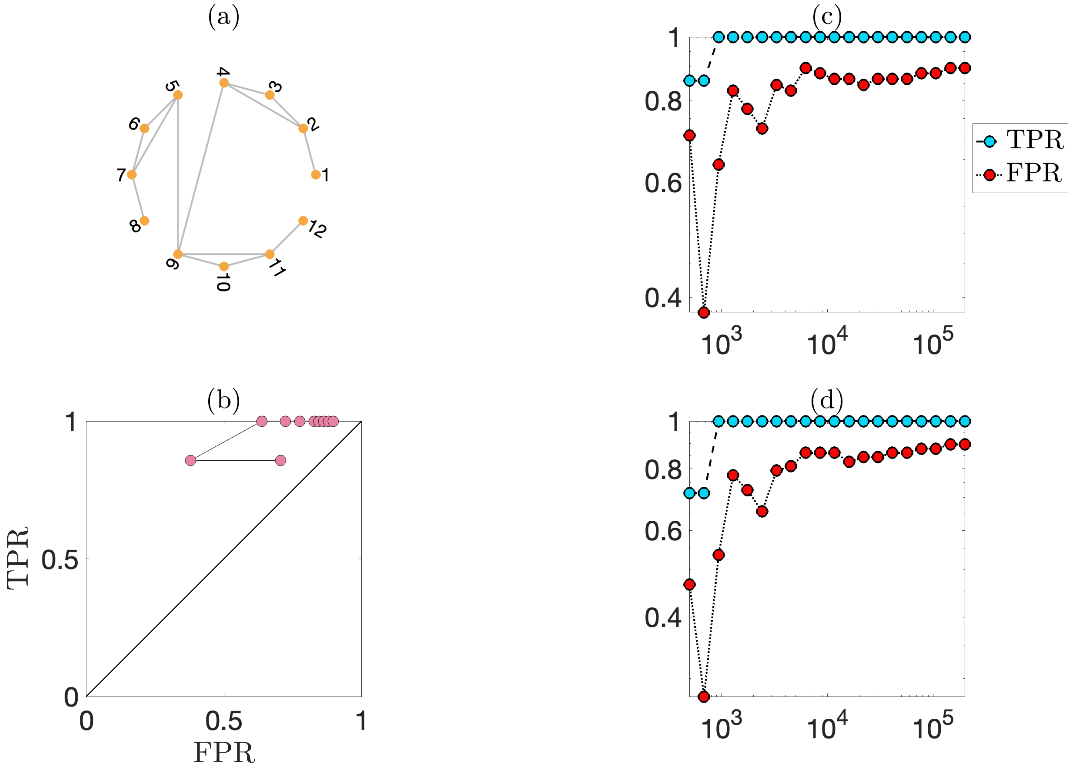

Figure 6.6: **Statistical tests using RSD in HR system.** Panels (a)-(d) present similar plots as in Fig. 6.2.

The methods using RSD (see Fig. 6.6) and TWSD(see Fig. 6.7) are implemented on the data at significance levels $\alpha = 0.1$ and $\alpha = 0.01$. Figure 6.6 shows the result of the method using RSD from the HR system. Panels (b) and (c) show the evolution of TPR and FPR as a function of time length at significance levels, $\alpha = 0.1$ and $\alpha = 0.01$, respectively. In both, the method using RSD performs poorly because of a high FPR exceeding 0.8. This implies that connectivity does not arise solely from amplitude correlation, as the method using RSD, lacking the amplitude correlation in the data, does not achieve perfect inference. Figure 6.6 (b) shows this poor performance of the method depicting the pairs of TPR, FPR on the plane, where the points are far from the perfect inference point at $(\mathrm{TPR}, \mathrm{FPR}) = (1, 0)$.

Figure 6.7 shows the results of the method using TWSD on the HR system. Panels (b) and (c) demonstrate that it also performs poorly in network inference at both $\alpha = 0.1$ and $\alpha = 0.01$ because of high FPR. Furthermore, Fig. 6.7(d) plots the pairs of $(\mathrm{TPR}, \mathrm{FPR})$ on

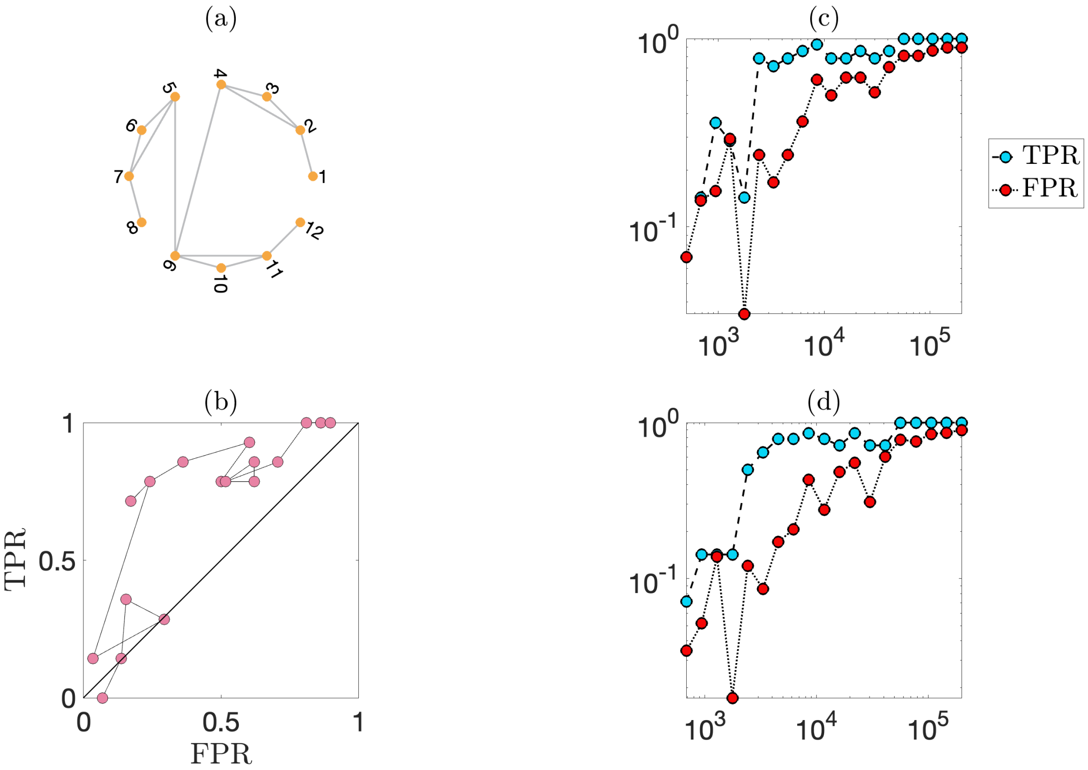

Figure 6.7: **Statistical tests using TWSD in HR system.** Panels (a)-(d) present similar plots as in Fig. 6.2.

the plane at $\alpha = 0.01$, which is also an indicator that the method performs poorly, as the points are far from perfect inference, TPR= 1 and FPR= 0. A high FPR indicates that the method infers extra links not present in the original network, implying that destroying phase synchronisation results in a significantly lower MIR between unconnected nodes and connected ones. In other words, unconnected nodes in the system exhibit relatively high phase synchronisation. In conclusion, the source of connectivity is not solely amplitude correlation or phase synchronisation.

6.2.4 Stochastic Dynamical Systems

6.2.4.1 Stochastic Kuramoto Oscillators

We numerically solved the Stochastic Kuramoto model in Eq. (3.18) using the adjacency matrix, **A** of the network in Fig. 6.8(a), using the Euler-Maruyama method following

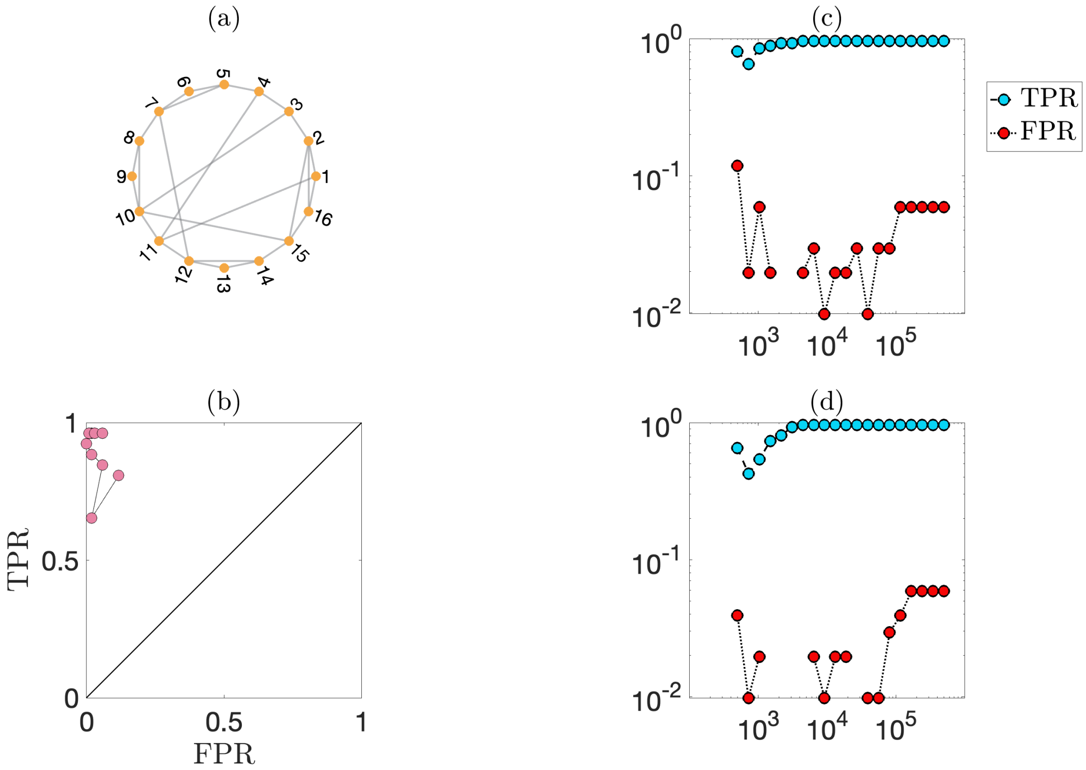

Figure 6.8: **Statistical tests using RSD in Stochastic Kuramoto.** Panels (a)-(d) present similar plots as in Fig. 6.2.

the algorithm 4 (see Sec.3.3.3), with a step size, $h = 0.05$, the final integration time, $t_f = 25\times10^3$, and stochastic strength, $D = 0.05$. The initial conditions for the θ_i are chosen uniformly random from the interval of $[0, 2\pi]$. The first half of the data was discarded, considering a transient period. Therefore, the last $250,000$ data points were used in the network inference. We used the instantaneous frequency of the data computed from Eq. (4.5) as a probe in network inference, implementing the method of statistical tests using both RSD and TWSD.

Fig. 6.8 shows the results of the method using RSD on the stochastic Kuramoto system. Panels (c) and (d) show the evolution of TPR and FPR as a function of time at significance levels, $\alpha = 0.1$ and $\alpha = 0.01$. Here, TPR eventually reaches to 1 although FPR stays at 0.06 in both, however, the significance level $\alpha = 0.01$ improves FPR for shorter time length. Consequently, the method of statistical tests using RSD cannot achieve the perfect inference, which is also demonstrated in Fig. 6.8(b).

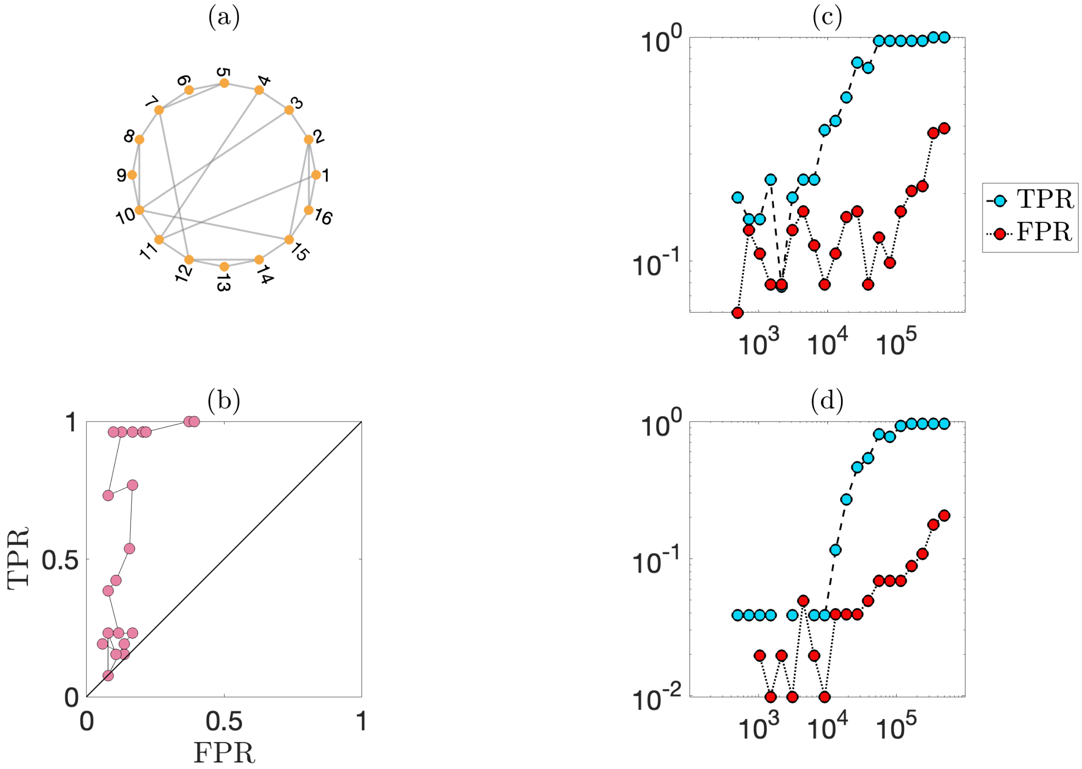

Figure 6.9: **Statistical tests using TWSD in Stochastic Kuramoto.** Panels (a)-(d) present similar plots as in Fig. 6.2.

We also implemented the method of statistical tests using TWSD on the data. In Fig. 6.9, panels (c) and (d) show the evolution of TPR and FPR as a function of time length at significance levels, $\alpha = 0.1$ and $\alpha = 0.01$. Here, the method does not achieve perfect inference due to high FPR. Taking into account the method at the significance level, $\alpha = 0.01$ improves its performance as FPR decreases from 0.39 to 0.21 for the full data set. Panel (b) shows the pairs of (TPR, FPR), which are distant from the perfect inference (TPR, FPR) = (1, 0). In conclusion, the statistical test using RSD and TWSD does not result in a perfect inference.

6.3 Summary

In this chapter, we discussed the method of hypothesis tests using RSD and TWSD for network inference as proposed by the author in [2]. The method is based on the comparison of MIR values of the pairs of nodes in the original data with those in the surrogate data. Using two surrogate data RSD and TWSD, we implemented the method of hypothesis tests on different systems, namely, Gaussian-correlated data, logistic map, circle map, HR system, Lorenz system, and stochastic Kuramoto phase oscillators.

	mean PC[1]	std. PC [1]	ρ	MLE	RSD[2]	TWSD[2]
Gaussian correlated	0.61	0.20	0.30	-	(1.00, 0.03)	(0.13, 0.12)
Logistic map	0.03	0.03	0.85	0.51	(1.00, 0.02)	(1.00, 0.34)
Circle map	0.01	0.004	0.83	1.33	(1.00, 0.00)	(1.00, 0.30)
HR system	0.26	0.18	0.68	0.41	(1.00, 0.90)	(1.00, 0.90)
Lorenz system	0.02	0.02	0.24	0.99	(0.96, 0.00)	(1.00, 0.91)
Stochastic Kuramoto	0.11	0.12	0.22	-	(0.96, 0.06)	(1.00, 0.39)

Table 6.1: Summary of the results.

[1] PC values corresponding p-values are less than 0.05, are considered.
[2] The pair of (TPR, FPR) is given.

Table 6.1 summarises the results of the network inference from the trajectories of the systems considered by presenting the pairs of (TPR, FPR) once the whole data set is considered at the significance level, $\alpha = 0.01$. It also provides the mean absolute PC and its standard deviation in pairs with statistically significant PC values (with values p less than 0.05), as well as the Kuramoto phase order, ρ, of the phases, and the MLE of the trajectory for the corresponding system. In the logistic and circle maps, although their phases are highly synchronised and the mean absolute PC is quite low, RSD works better than TWSD. It is possible to give other examples from the results in Table 6.1. These results imply that the level of mean PC or ρ does not give any information about the source of connectivity. Finally, we can state two drawbacks of the method of hypothesis tests using RSD or TWSD:

1. RSD and TWSD destroys all amplitude correlation or phase synchronisation among the variables, respectively. However, the unconnected pairs of nodes might have amplitude correlation or phase synchronisation even at a lower level than the connected pairs because they are connected through another node. Therefore, a method that does not destroy all amplitude or phase relation at once but gradually, might be more successful in network inference.

2. Identifying the source of connectivity in a network is a subtle task. Their global phase order ρ and mean absolute PC do not give any information about the source of connectivity. Therefore, we do not have any sign to select appropriate surrogate data.

To overcome the drawbacks, we introduce the APMSD method in the next chapter. It allows us to investigate the source of connectivity and infer network successfully for all systems presented here. By doing that, APMSD entails foregoing advantage of the present method that is applicable to time series data without prior knowledge of its network.

Chapter 7

Network Inference using APMSD

In the previous chapter, we studied the method of hypothesis tests with RSD and TWSD implemented on various network and dynamics topologies and discussed its pros and cons. We propose the APMSD method to address the two main drawbacks: (1) to identify the source of connectivity as a contribution of instantaneous amplitudes and phases; (2) to infer the network structure considering different levels of amplitude correlation and phase synchronisation by destroying them gradually instead of all at once, as in the use of RSD and TWSD.

This chapter is dedicated to introducing the method with its theoretical background and to presenting the results from different dynamics and network topologies.

7.1 Theory of APMSD Method

Here, we introduce a new approach to generate surrogate data sets for network inference, which we call the APMSD method. Because connectivity can emerge through the contribution of amplitude correlation and/or phase synchronisation, which is a priori unknown, our method transforms a signal $x(t)$ into the analytic signal, $z(t)$, introduced in 1946 by Gabor [23], by using the HT (see Sec. 4.3.3). This allows us to compute the instantaneous amplitude, $A(t)$, and instantaneous phase, $\phi(t)$, of the real signal, $x(t)$. By introducing a percentage of randomisation pc_1 of the instantaneous amplitude, $A(t)$, and pc_2 of the instantaneous phases, $\phi(t)$, we can generate a surrogate data set for $x(t)$ by computing the real part of $z'(t)$, where $z'(t)$ is the analytic signal of the randomised instantaneous amplitude, $A'(t)$ and the instantaneous phases, $\phi'(t)$.

In particular, if $x(t)$ is the real signal of time recordings of a variable X, one can compute its analytic signal, $z(t)$ and its components, the instantaneous amplitudes, $A(t)$ and the instantaneous phases, $\phi(t)$ using Eq.(4.9).

Next, we compute the randomised instantaneous amplitudes and phases, $A'_{pc_1}(t)$ and $\phi'_{pc_2}(t)$, respectively, using the percentages, pc_1, of randomisation of instantaneous amplitudes and pc_2 percent of randomisation of instantaneous phases. For instance, $pc_1 = 10$ and $pc_2 = 50$ mean that we replace 10% of instantaneous amplitudes with uniformly selected random numbers in x_r^A and 50% of the instantaneous phases with uniformly selected random numbers in x_r^ϕ to obtain randomised instantaneous amplitude $A'(t)$ and randomised instantaneous phases $\phi'(t)$. In this context, x_r^A and x_r^ϕ are the intervals of real numbers between the minimum and maximum values in $A(t)$ and $\phi(t)$ of all N recordings $x(t)$, respectively. In particular, we do this randomly choosing pc_1 percent of $A(t)$ and pc_2 percent of $\phi(t)$, and replacing them with the same number of uniformly random numbers in the intervals x_r^A and x_r^ϕ, respectively, resulting in $A'_{pc_1}(t)$ and $\phi'_{pc_2}(t)$. Finally, we compute a surrogate data set, $x'(t)$, of $x(t)$ by computing

$$x'(t) = A'_{pc_1}(t)\cos(\phi'_{pc_2}(t)), \tag{7.1}$$

which is the real part of the randomised analytic signal $z'(t)$ (see Eq. (4.9)). Algorithm 6 summarises the steps in the computation of APMSD.

After generating APMSD at any pair of randomisation percentage, (pc_1, pc_2) we implement hypothesis testing as discussed in Chp. 6 (see algorithm 5). The result of hypothesis tests using APMSD for any pair of percentages, pc_1 and pc_2, is an inferred adjacency matrix, $\tilde{\mathbf{A}}$. By varying pc_1 and pc_2 in $[0, 100]$ and comparing the resulting $\tilde{\mathbf{A}}$ with the original adjacency matrix, $\mathbf{A}$, one can identify the best pairs that result in successful network inference, that is, in $\tilde{\mathbf{A}} = \mathbf{A}$.

Algorithm 6 Algorithm for APMSD

```
Input signal $x(t)$
Input randomisation parameters $pc_1$ and $pc_2$
Obtain analytic signal, $z(t)$ from $x(t)$
Compute $A'_{pc_1}$                  ▷ $A'_{pc_1}$ is $pc_1\%$ randomised of original amplitude
Compute $\phi'_{pc_2}$                ▷ $\phi'_{pc_2}$ is $pc_2\%$ randomised of original phases
$z'(t) = A'_{pc_1} e^{i\phi'_{pc_2}}$                        ▷ Randomised analytic signal
$x'(t) = A'_{pc1}(t)\cos(\phi'_{pc2}(t))$                 ▷ Randomised real-valued signal
Return $x'(t)$
```

Comparing the inferred adjacency matrix $\tilde{\mathbf{A}}$ with the original adjacency matrix $\mathbf{A}$, we compute TPR and FPR for any pair of percentages, (pc_1, pc_2) in $[0, 100] \times [0, 100]$ (see Eq.5.18). We then partition this space into a grid 11×11, which results in 121 pairs (pc_1, pc_2) at the nodes of the grid. For the entire data set, we represent the values of TPR and FPR on this parameter space. In particular, TPR values represented by the colour map (see, for example, the colour map on the panel (c) in Fig. 7.1) and the presence of FPR is shown by the red circle whose radius is proportional to FPR the value in the given cell. If the pairs of randomisation parameters lead to perfect inference, which mean TPR $= 1$ and FPR $= 0$, the green circle indicates the corresponding cell.

Finally, for any time length of the data set, we compute the ROC distance (see Eq. (5.19)) over the parameter space and plot the pair of $(\mathrm{TPR}, \mathrm{FPR})$ that corresponds to the minimum distance. This plot is useful because it gives us the minimum time length of the data set in which the method can achieve perfect inference in the sense that $\tilde{\mathbf{A}} = \mathbf{A}$ (see, for example, panel (b) in Fig. 7.1).

7.2 Results

7.2.1 Gaussian Distributed Correlated Data

Here, we present the result of the implementation of the method of hypothesis tests using APMSD on the same data as in Sec. 6.2.1. Assuming that there is no connection between nodes 8 and 9 due to the relatively small PC between them, 0.23, the bar graph shows

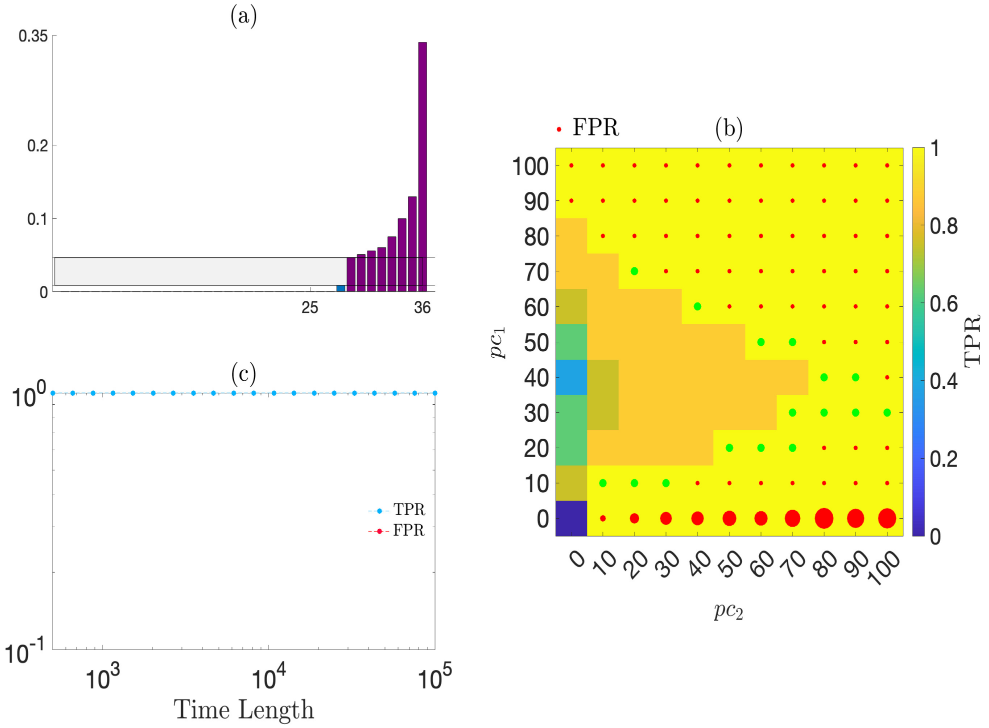

Figure 7.1: **Statistical tests using APMSD on Gaussian-distributed data.** (a) The plot of the ordered MIRs, where purple bars represent the connected pairs, blue bars unconnected ones. The grey stripe classifies into connected and unconnected pairs of nodes. (b) Plot of parameter space of pc_1 and pc_2, where the colour map illustrates the TPR, with red circles indicating the amount of FPR (size proportional), green circles highlight pairs of parameters leading to correct network inference. (c) The evolution of TPR and FPR over time length is presented. The method requires at least 500 data points to find at least one pairs of randomisation parameters, pc_1, pc_2 resulting in the perfect inference.

that it is possible to identify a range of thresholds that lead to perfect inference, indicated by the grey stripe in Fig. 7.1(a). Here, the purple bars indicate MIR values of connected pairs and the blue bars MIRs of the unconnected pairs.

In Fig. 7.1(b), the colour map shows the values of TPR of the network inference for the corresponding pairs of randomisation parameters, (pc_1, pc_2), while the red circle represents the presence of FPR in any cell of parameter space, with its radius proportional to the values of FPR. To highlight perfect inference in any cell of the parameter space, in other words, the pairs of randomisation parameters satisfy TPR $= 1$ and FPR $= 0$, we used the green circle. The x axis represents the randomisation parameters of instantaneous phases,

pc_2, and the y axis represents the randomisation parameters of instantaneous phases, pc_1. Figure 7.1(b) shows that solely fully phase randomised surrogate data (corresponds to TWSD for $pc_1 = 0$ and $pc_2 = 100$) or fully amplitude and phase randomised surrogate data (which corresponds to RSD for $pc_1 = 100$ and $pc_2 = 100$) cannot achieve successful network inference as expected, since RSD and TWSD did not lead to perfect inference in the previous chapter. However, hypothesis tests using APMSD achieve successful network inference for some pairs of randomisation parameters that the green circles indicate.

For any time length, we can build a parameter space in the same way. Then, we can find the best pairs of $(\mathrm{TPR}, \mathrm{FPR})$ that give the minimum ROC distance (see Eq. (5.19)). Figure 7.1(c) shows that there is at least one pair of randomisation parameters, (pc_1, pc_2), which leads to perfect network inference for all the time lengths considered. It is important to note that the y -axis is on the logarithmic scale, so the FPR value does not appear if it is equal to 0.

7.2.2 Deterministic Discrete Systems

7.2.2.1 Logistic Map

Here, we present the result of the implementation of the method of hypothesis tests using APMSD on the same data as in Sec. 6.2.2.1. As shown in Fig. 7.2(a), there is a range of threshold values that leads to the perfect inference. However, in the upper right corner (corresponds to RSD) and in the lower right corner (corresponds to TWSD) in the parameter space, we cannot infer the network successfully, as shown in Fig. 7.2 (b), which is in agreement with our findings in Table 6.1. However, hypothesis tests using APMSD achieve perfect inference for the parameters of the randomisation percentages, (pc_1, pc_2) shown by green circles. Figure 7.2(c) demonstrates how the pairs of $(\mathrm{TPR}, \mathrm{FPR})$ change over time length; at least 3×10^4 data points are needed to infer the network structure correctly.

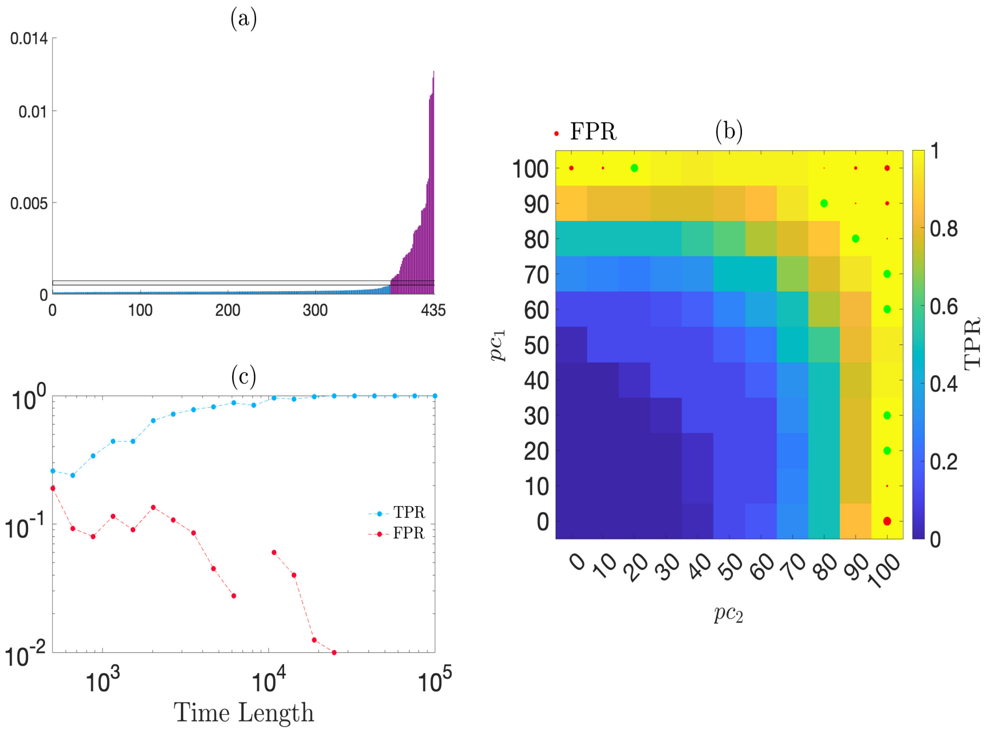

Figure 7.2: **Statistical tests using APMSD on the coupled logistic map.** Panels (a) plots the ordered MIR values, where the grey stripe classifies into connected (indicated by purple bars) and unconnected (indicated by blue bars) pairs of nodes, (b) shows the plot of parameter space of pc_1 and pc_2, where the colour map illustrates the TPR, with red circles indicating the amount of FPR (size proportional), green circles highlight pairs of parameters leading to correct network inference, and (c) plots the evolution of TPR and FPR over time length is presented. The method requires at least 3×10^4 data points to find at least one pairs of randomisation parameters, pc_1, pc_2 resulting in perfect inference.

7.2.2.2 Circle Map

Here, we present the result of the implementation of the method of hypothesis tests using APMSD on the same data as in Sec. 6.2.2.2. As shown in Fig. 7.3(a), it is possible to find a range of thresholds leading to perfect inference. When we consider the full data length, while the randomisation parameters in the upper right corner of the parameter space (corresponds to RSD) achieve the perfect inference, the randomisation parameters in the lower right corner of the parameter space (corresponds to TWSD) do not allow inferring the original network successfully, as shown in Fig. 7.3(b). These results agree with our findings in Table 6.1, that RSD achieves the network inference successfully but TWSD do not. However, the hypothesis test using APMSD achieves perfect inference for some parameters of randomisation percentages, (pc_1, pc_2) shown by the green circles. Figure 7.3 (c) demonstrates the change in the pairs of $(\mathrm{TPR}, \mathrm{FPR})$ over the time length; at least 10^4 data points are needed to infer the network structure correctly.

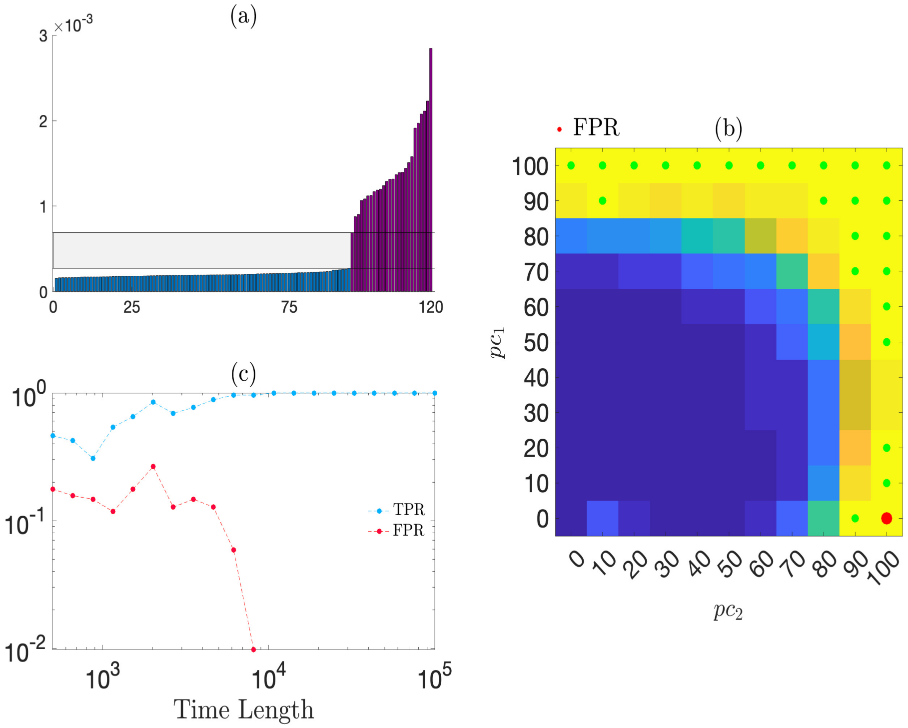

Figure 7.3: **Statistical tests using APMSD on the coupled circle map.** Panels (a)-(c) present comparable plots as in Fig. 7.2.

Besides the dynamics, network size and its structure, one of the important parameter affects the performance of network inference is the network density (for density see Sec. 2.2.1). We used networks of 12 nodes with varying numbers of links to test the proposed method against network density (see Fig. 7.4). The data sets are generated from the coupled circle maps based on the connectivity of 12 nodes with a varying number of links, which creates the different density given on the y-axis. We also used different time duration of the data sets to observe the performance change through it, as shown on the x-axis. The minimum ROC distance for each network and time length that is obtained using the best pair of randomisation parameters, represented by the colour map, where the dark blue region stands for the perfect inference i.e. the ROC distance (see 5.19) equals zero. The method of hypothesis tests using APMSD can achieve inference for sparse networks for the shorter time-series length, whereas it requires longer time series to infer the denser network successfully.

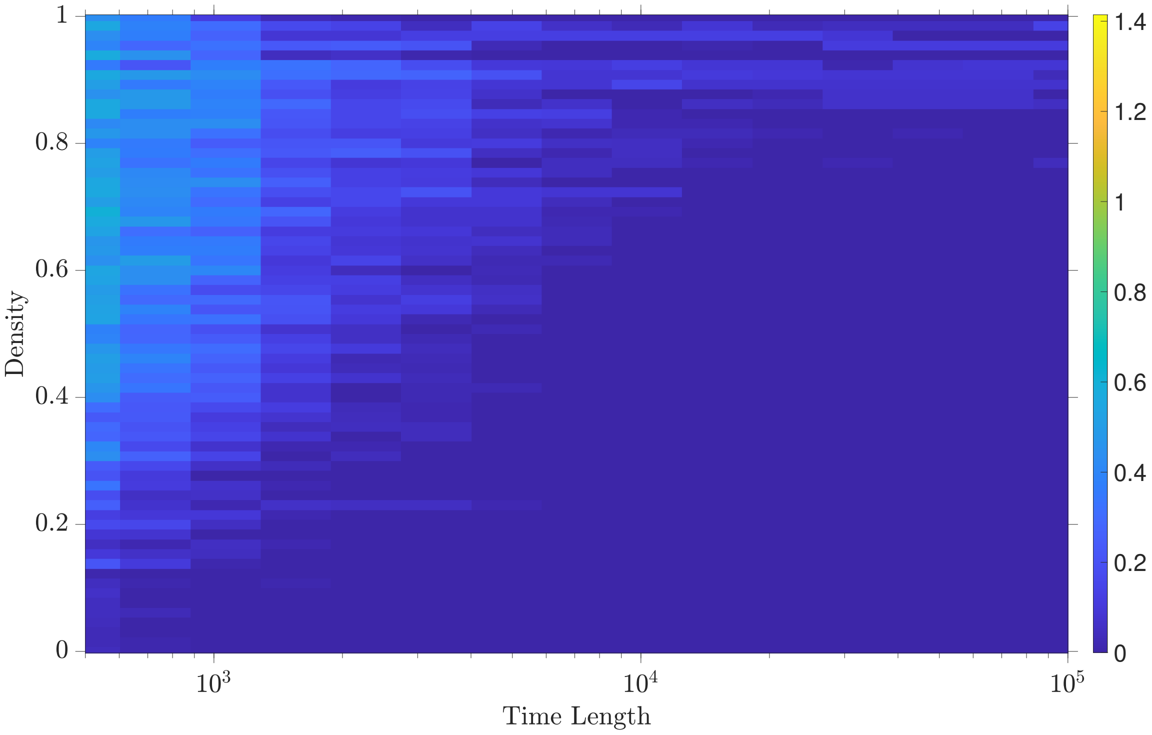

Figure 7.4: **Density of network vs time-series length.** We used networks of 12 nodes with varying number of links leading to different level of density. Colour map shows the ROC distance of network inference for the corresponding time length and density.

7.2.3 Deterministic Continuous Systems

7.2.3.1 Lorenz System

Here, we present the result of the implementation of the method of hypothesis tests using APMSD on the same data as in Sec. 6.2.3.1. When considering the whole length of the data, it is possible to find a range of thresholds leading to perfect inference, as shown in Fig. 7.5(a). When we randomise all instantaneous amplitudes and almost all instantaneous phases, the method can infer the network successfully, as shown in Fig. 7.5(b) by the green circles. Interestingly, the method of hypothesis tests using RSD cannot achieve perfect

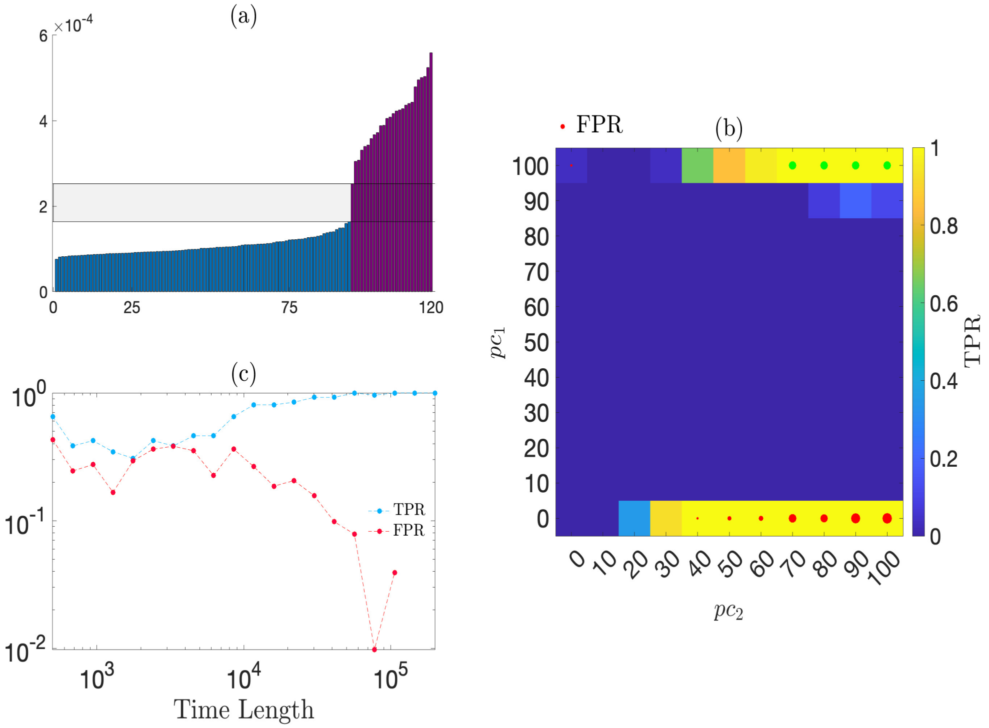

Figure 7.5: **Statistical tests using APMSD on the coupled Lorenz system.** Panels (a)-(c) depict comparable plots as in Fig. 7.2.

inference with TPR = 0.96 and FPR = 0 (see Table 6.1) but the parameters leading to fully randomised amplitudes and phases (shown in the upper right corner for the pair of parameters $(pc_1, pc_2) = (100, 100)$) infer the network correctly. This difference might come from the different randomisation processes; RSD randomises all the signals directly, while, in this method, we compute the randomised signal from the randomised amplitude and phases using Eq. (7.1). Furthermore, the cell in the lower right corner of the parameter space, i.e. $(pc_1, pc_2) = (0, 100)$, shows the result of $(\mathrm{TPR}, \mathrm{FPR}) = (1, 0.16)$, randomising all phases similar to the TWSD whose result of network inference is $(\mathrm{TPR}, \mathrm{FPR}) = (1, 0.90)$ (see Table 6.1). The difference in the phase randomisation process between this method and TWSD is that the first is based on the HT while the latter randomises phases through the FT in the frequency domain (see 6.1.2.2). This difference causes a significant decrease in FPR. Figure 7.5 (c) demonstrates the change in the pairs of $(\mathrm{TPR}, \mathrm{FPR})$ over the time length; at least 1.5×10^5 data points are required to infer the network structure correctly.

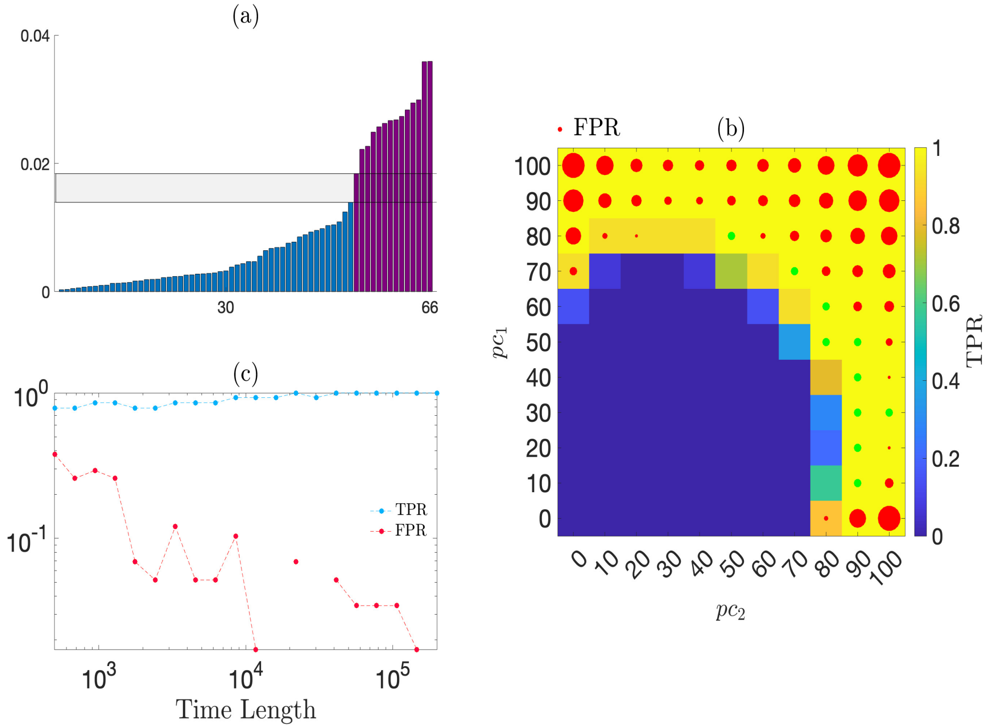

Figure 7.6: **Statistical tests using APMSD on the HR system.** Panels (a)-(c) depict comparable plots as in Fig. 7.2.

7.2.3.2 Hindmarsh-Rose Model

Here, we present the result of the implementation of the method of hypothesis tests using APMSD on the same data as in Sec. 6.2.3.2. When considering the whole length of the data, it is possible to find a range of thresholds leading to perfect inference, as shown in Fig. 7.6(a). When we randomise all instantaneous amplitudes and phases (or only all instantaneous phases), corresponding to the RSD (or the TWSD), the method of hypothesis tests with APMSD gives high FPR as expected in Table 6.1 (see Fig. 7.6(b)). There are some pairs of randomisation parameters, (pc_1, pc_2), which are shown by green circles, which lead to inferring the network structure correctly. Thus, we conclude that the source of connectivity comes from the combination of instantaneous amplitudes and phases. Figure 7.6(c) show the TPR and FPR as a function of time length: we require at least 2×10^5 data points to have perfect inference.

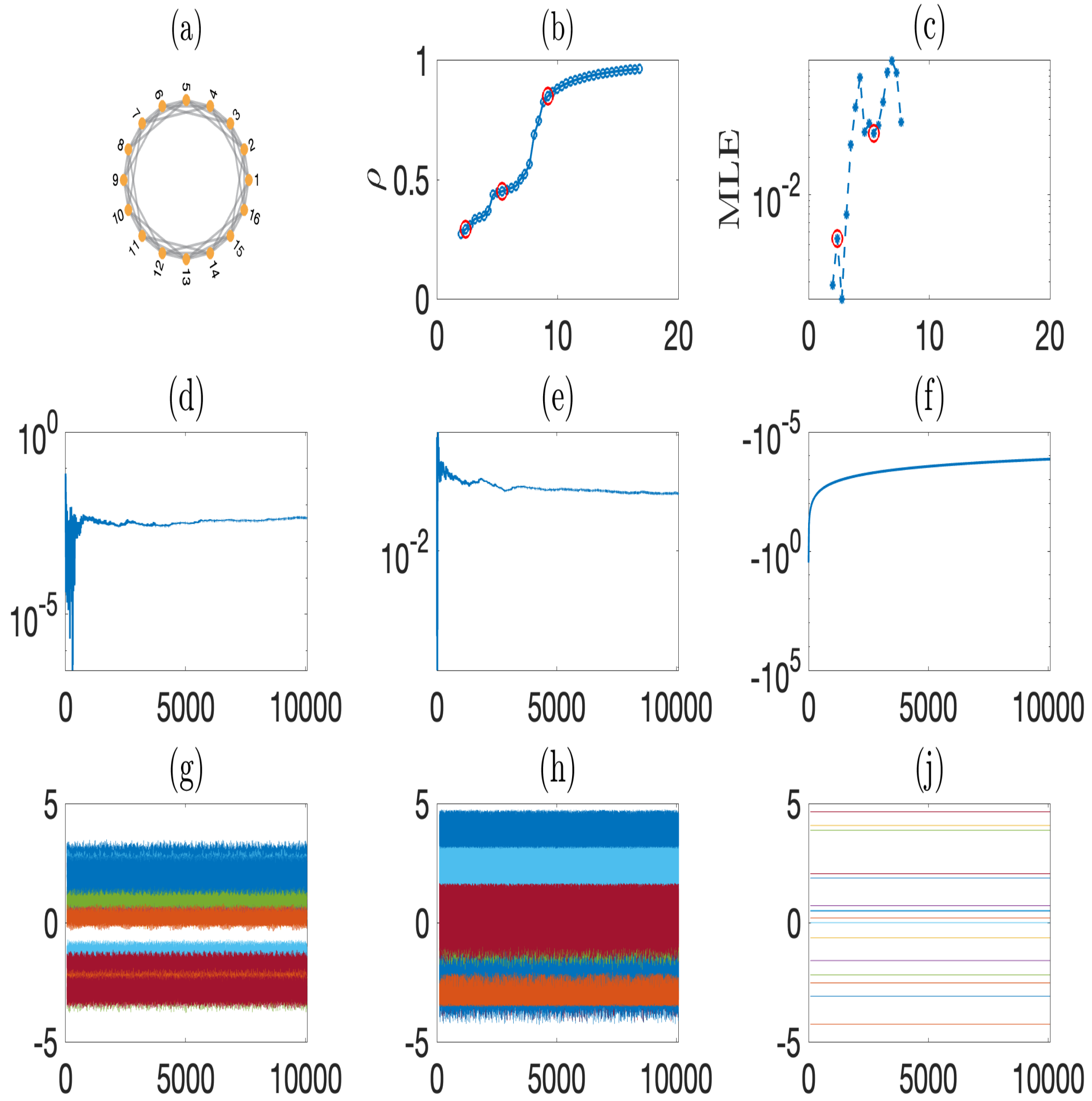

Figure 7.7: **Three trajectories of the coupled deterministic Kuramoto oscillators.** The trajectories generated based on the connectivity of 6-RL of 16 nodes in (a). Kuramoto order of the trajectories, ρs, increases as coupling strength increases in (b). We choose three trajectories having different level of ρ, shown by red circles. MLEs of the trajectories have been shown, and the selected trajectories are pointed out by red circles in (c). As MLE is presented in log-scale and third trajectory is not chaotic, there are only two circles appear in the plot. (d)-(f) show MLEs as a function of time (g)-(j) show the instantaneous frequencies of the trajectories for 16 nodes, where the non-chaotic trajectory results in constant frequencies.

7.2.3.3 Deterministic Kuramoto Oscillators

In this section, we investigate the effect of phase synchronisation on the performance of our approach. To do that, we implement the method of hypothesis test using APMSD on three trajectories generated from deterministic Kuramoto phase oscillators. In this sense, based on connectivity in the 6-RL of 16 nodes (see Fig. 7.7(a) and Sec. 2.3), we generated the trajectories with 10^5 data points and 0.1 step size by the coupled deterministic Kuramoto oscillators, using Eq. (3.17), for the coupling strength, K ranging in $[2, 16.7]$. The Kuramoto phase order, ρ, tends to increase as K increases. We choose three trajectories with different levels of ρ, to test our approach in different levels of phase synchronisation, are shown by red circles in panel (b) and their MLEs are shown in panel (c) in Fig. 7.7. As the MLEs are presented in log-scale and the third trajectory is not chaotic, we see only two red circles in this figure. Panels (d)-(e) show MLEs as a function of time; the first two are chaotic, but the third is a non-chaotic trajectory. As we discussed previously in Sec. 3.3.2.3, instead of phases, we used instantaneous frequencies (as shown in Fig. 7.7(g)-(j)) as probes to infer the network.

It is impossible to assign a threshold clustering the MIR values in two groups as connected and unconnected in all cases (see Fig. 7.8(a), (c), (e)). Therefore, our approach cannot provide any pairs of randomisation parameters (pc_1, pc_2) to infer the network successfully (see Fig. 7.8(b), (d), (f)). The phase synchronisation of the trajectories increases from left to right, and we observe that both TPR and FPR of our approach increase as ρ increases.

7.2.4 Stochastic Dynamical Systems

7.2.4.1 Stochastic Kuramoto Oscillators

In this section, we present the results of our approach on the stochastic Kuramoto phase oscillators in two parts. Firstly, we employ the method on the trajectory used in Sec. 6.2.4.1. Recall that we cannot successfully infer the network using hypothesis tests using RSD and TWSD. In the second part, our aim is to investigate the effect of the stochastic

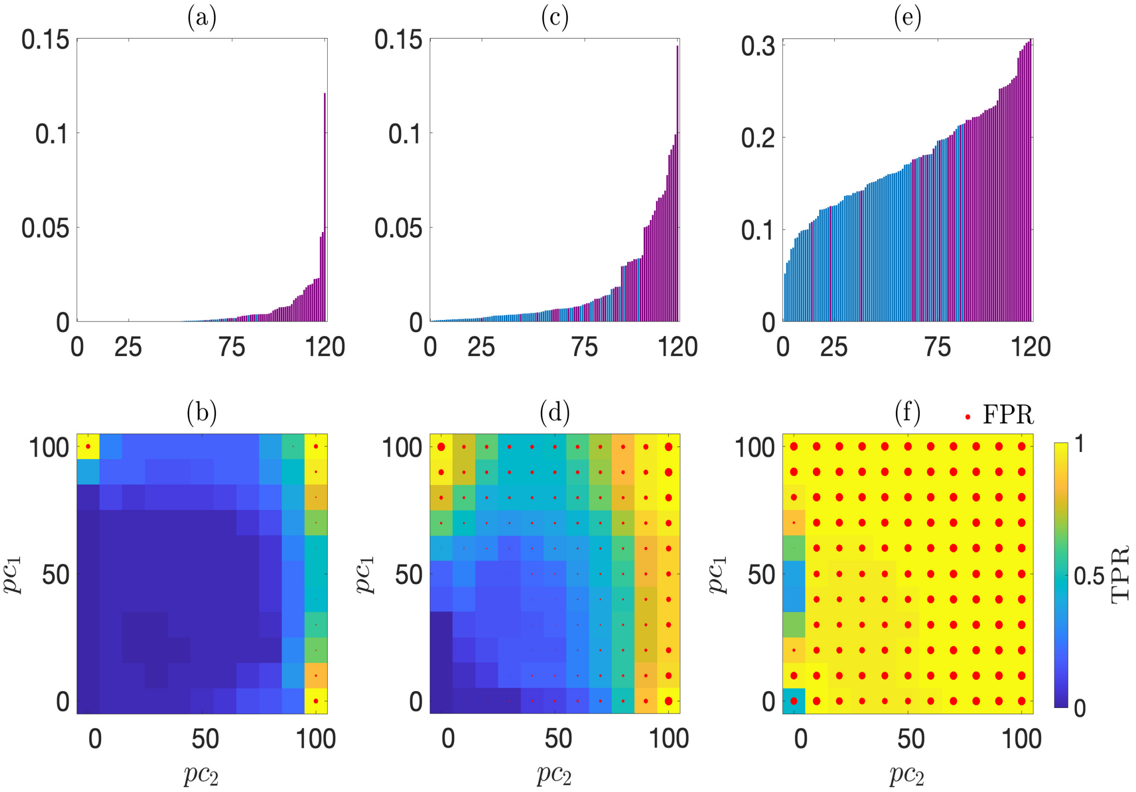

Figure 7.8: **Statistical tests using APMSD on the instantaneous frequencies of three trajectories generated by deterministic Kuramoto oscillators.** Kuramoto phase order, ρ, increases from left to right. Panels (a), (c), and (e) show that there is no threshold value that gives perfect inference. Panels (b), (d) and (f) show the results of the statistical test using APMSD in three trajectories. There are no pairs of randomisation parameters, (pc_1, pc_2), which allows perfect inference. Both TPR and FPR increases as ρ rises.

strength, D, network types, and time length on the performance of the method. In this sense, we used the network of 16 nodes and 26 links, Watts-Strogatz small-world network, Erdős-Rényi random network and Barabási-Albert scale free network of 30 nodes with comparable density.

Figure 7.9(a) shows that there is no threshold separating connected pairs from unconnected ones, so thresholding cannot achieve perfect inference for the whole data. As hypothesis tests using RSD and TWSD cannot infer the network successfully, the corresponding randomisation parameters, $(pc_1, pc_2) = (100, 100)$ to RSD and $(pc_1, pc_2) = (0, 100)$ to TWSD do not allow the network to be inferred successfully due to FPR. However, there is a pair of randomisation parameters, $(pc_1, pc_2) = (90, 100)$, leads to the successful inference as shown by the green circle in Fig. 7.9(b). The values of TPR and FPR in log-scale as a function of time length are presented in Fig. 7.9(c) so FPR $= 0$ is absent. The method requires at least 2.5×10^4 data points to find at least one pair of randomisation parameter, (pc_1, pc_2), leading to perfect inference meaning that TPR $= 1$ and FPR $= 0$.

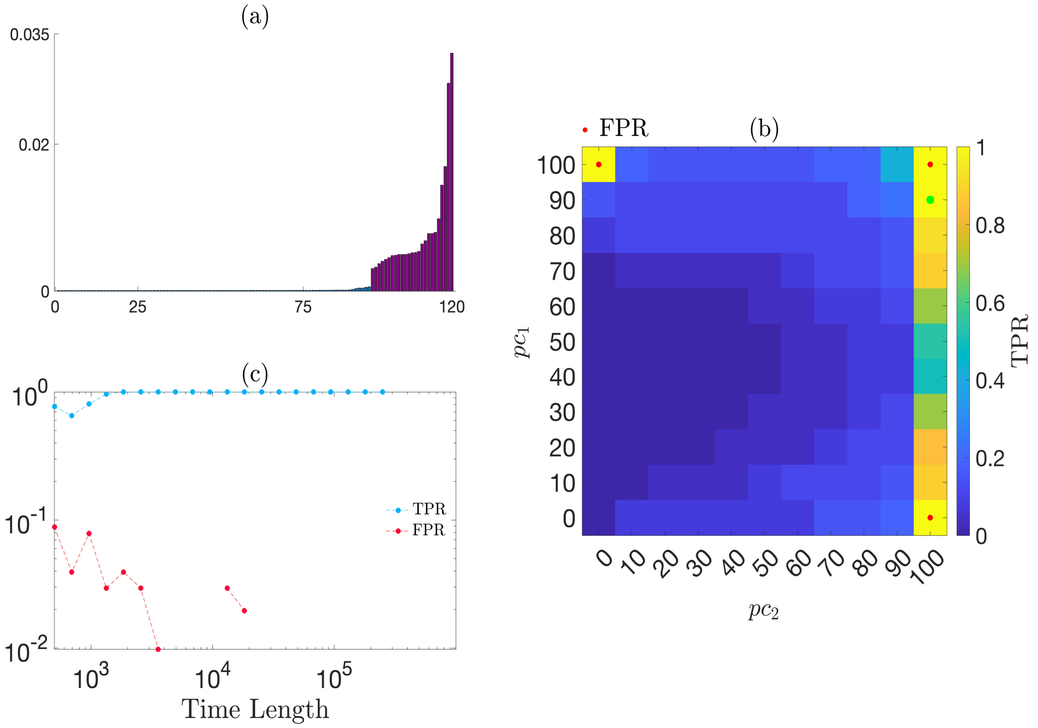

Figure 7.9: **Results of the stochastic Kuramoto system.** Panels (a)-(c) depict comparable plots as in Fig. 7.2.

We presented the results of a series of simulations of our approach on different types of networks to assess the impact of stochastic strength D, and time length based on network types (see Fig. 7.10). We expect to see that the method performs poorly as the randomness of the system, D, increases. From the simulations so far, we also expect to see better inference performance as time length increases and that there is not much difference among the network types with the same size and comparable density (see Sec. 5.4).

Figure 7.10 shows that the ROC distance for each pair of time length and stochastic strength, D, where dark blue corresponds to perfect inference with zero ROC distance, and yellow stands for the worst case with (TPR, FPR) $= (0, 1)$ the corresponding ROC distance

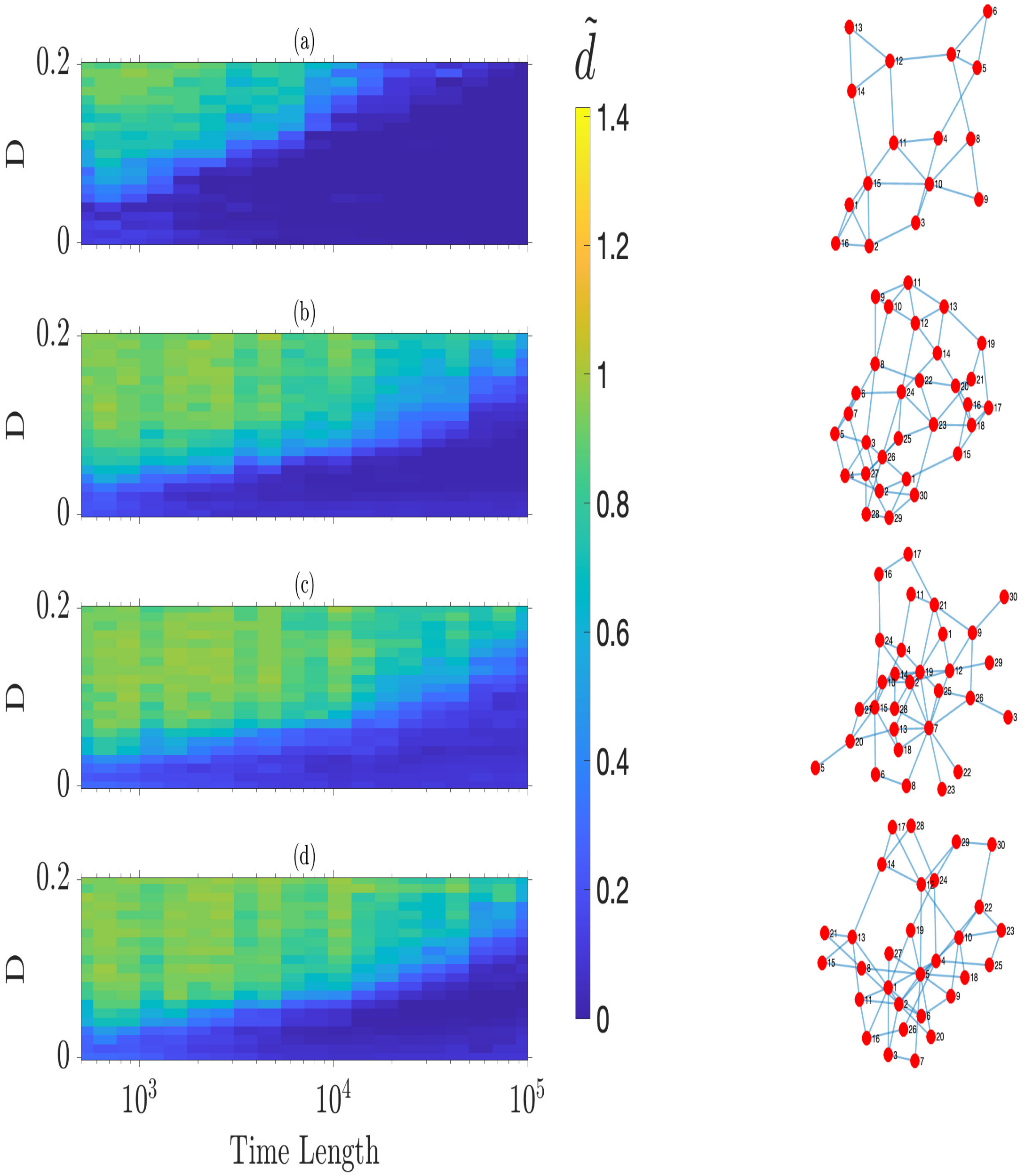

Figure 7.10: **Statistical tests using APMSD on the stochastic Kuramoto over parameter D and time-series length.** For each D and time length, the APMSD method produces 121 pairs of TPR, FPR in an 11×11 parameter space. The colours indicate the minimum Euclidean distance of these pairs to $(1, 0)$ representing perfect inference. The results for different network topology are presented in panels, (a) 16 nodes with 26 links (b) Watts-Strogatz small world network with 30 nodes and 60 links (c) Erdös-Renyi random network with 30 nodes and 55 links (d) Barabási-Albert scale-free network with 30 nodes and 57 links.

is $\sqrt{2}$. In all cases of network types, we see that a higher stochastic strength, D, requires a longer time to infer the network correctly. In particular, in the case of the network of 16 nodes (see Fig. 7.10(a)), at least 5×10^3 data points are required for $D < 0.1$, but it is necessary to have a longer time length for $D > 0.1$ to infer network successfully. For the highest D considered, the 10^5 data points allow the method to infer the network correctly.

Results from all other network types (see Fig. 7.10(b)-(d)), namely, Watts-Strogatz small world, Erdős-Rényi random and Barabási-Albert scale free networks, respectively, show similar pattern in the sense of network inference performance since they have the same number of nodes, $N = 30$ and comparable density. This supports our findings in Sec. 5.4. In all, for $D = 0.2$, the longest time length considered, 10^5 data points, are not enough to infer the network correctly; however, for $D < 0.1$, the ROC distance becomes very close to zero or at zero. Therefore, we can conclude that there are three important factors in network inference: (1) the network size affects the performance of the method, and we can see that it shows higher performance in the smaller networks by comparing the results from the network of 16 nodes and those from the network of 30 nodes. (2) Stochastic strength, D, has an effect on the dynamics of the system; higher D results in higher randomness and decreases the performance of network inference. (3) Time length is another factor that affects the performance of network inference. In general, longer time length helps MIR to be able to capture the interactions among nodes, especially in the presence of randomness.

7.2.5 Heterogeneous Network

So far, we have been studying the network with binary adjacency matrix but in this section, we aim to test our approach on the weighted adjacency matrix. Following [57], we obtain the weighted adjacency matrix,

$$\mathbf{W}_{ij} = \mathbf{A}_{ij}(1 + g\xi_{ij}), \tag{7.2}$$

where $\mathbf{A}$ is a binary adjacency matrix with the entries 0s and 1s, g is the stochastic strength of weighted adjacency matrix, and ξ is M-by-M matrix of a network of M nodes, whose

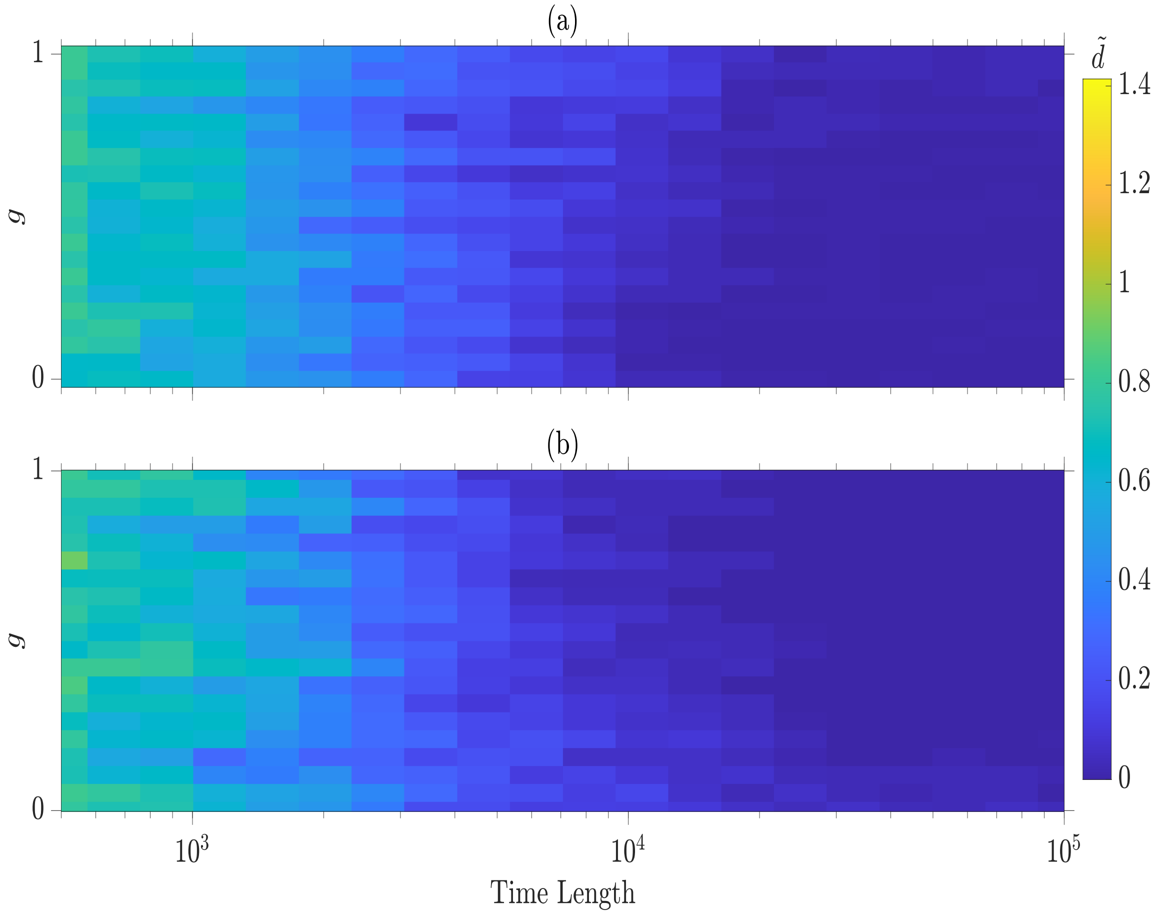

Figure 7.11: **Statistical test using APMSD on the heterogeneous dynamics and network.** The plots show the minimum Euclidean distance of the pair of (TPR, FPR) to the (1, 0) indicative of perfect inference, $\tilde{d}$, for network types (a) Watts-Strogatz small world network of 16 nodes and 32 links, (b) Erdös-Rényi random network of 16 nodes and 32 links.

entries are randomly selected from the interval $[-1, 1]$. We previously used Eq. (3.13) to generate data from coupled maps based on connectivity in the adjacency matrix $\mathbf{A}$. Replace $\mathbf{A}$ with $\mathbf{W}$ in Eq. (3.13), one can obtain the coupled map equation based on the weighted connectivity in $\mathbf{W}$ by,

$$x_{n+1}^i = (1-\alpha)f(r_i, x_n^i) + \frac{\alpha}{k_i}\sum_{j=1}^{M} \mathbf{W}_{ij} f(r_j, x_n^j) \tag{7.3}$$

The system we study in this section also differs from the previous ones in the sense of dynamics. The dynamics of the nodes of a network in previous simulations were identical with the same model and parameters; however, we generate the data from the system of the coupled circle maps using a randomly assigned parameter, r in $[0.01, 0.35]$, for each node in the network, resulting in nonidentical dynamics of nodes.

We used the adjacency matrix of the Watts-Strogatz small world and Erdős-Rényi random networks of 16 nodes and 32 links. After getting the weighted adjacency matrix, $\mathbf{W}$, from Eq. (7.2), we generate the data using the coupled circle map in Eq. (3.12) as f in Eq. (7.3) with coupling strength, $\alpha = 0.1$.

Figure 7.11 shows the ROC distance (see Eq. (5.19)) by colour ranges from dark blue to yellow, where dark blue represents the zero ROC distance, in other words, perfect inference for two different network structures, Watts-Strogatz small-world network on panel (a) and Erdős-Rényi random network on panel (b), both with 16 nodes and 32 links. They show similar performance in the sense of time length and stochastic strength of the weighted network g. At least 2×10^4 data points are required to infer the network structure correctly for most of the g values. Furthermore, no clear effect of g on network inference is observed.

7.3 Computational Aspect

Although our approach provides a new perspective to infer network structure and be able to observe contributions of amplitude correlation and phase synchronisation, there are two main drawbacks of it: (1) If the network structure is a priori unknown, we do not have any clue about the best pairs of randomisation parameters, (pc_1, pc_2). (2) Since we consider all parameter space and it increases the computational cost. For example, we run our simulation in a parameter space 11×11, which means that we repeat the hypothesis tests 121 times for each pair of (pc_1, pc_2).

We have two main factors influencing computational cost: (1) network size and (2) time length. Figure 7.12 shows the relation of computational time with the number of nodes, M, the number of pairs, $\frac{M(M-1)}{2}$ and the time length, L, for some simulations of the coupled logistic maps. It shows that the 2^{nd} degree polynomial regression fits well to the computational time as a function of the number of nodes, but the linear regression fits well to it as functions of the number of pairs and the time length. To obtain a single equation

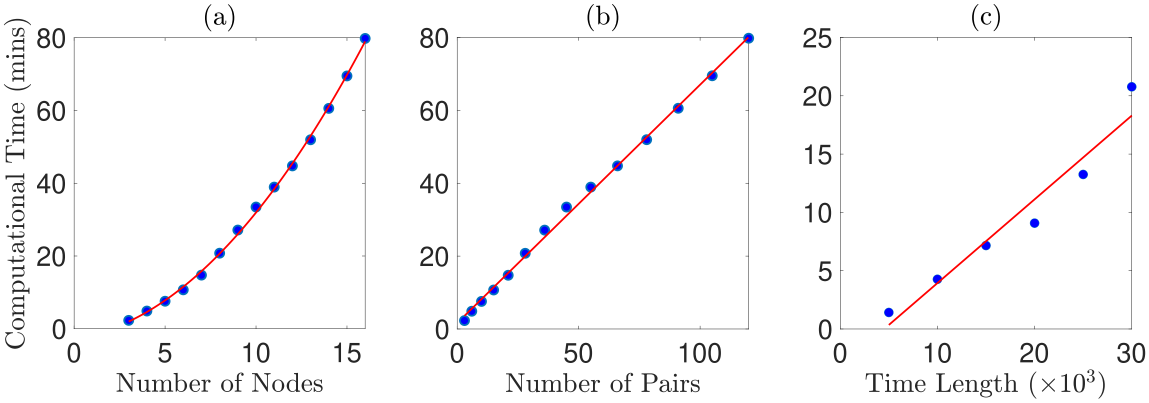

Figure 7.12: **Computational time** as a function of (a) number of nodes, (b) number of pairs, (c) and time length.

	Estimate	**SE**	**test-statistics**	**p**
Intercept	-22.5	1.22	-18.51	3.141e-12
Number of Pairs	0.66	0.01	63.006	1.3265e-20
Time Length	0.79	0.045	17.27	9.0957e-12

Table 7.1: Model output of multiple linear regression.

involving these two factors to estimate computational time, we conducted a multiple linear regression using the values of number of pairs and time length as independent variables and computational time as response variable. The model output is given in Table 7.1. R^2 value of the model is 0.99 meaning that the 99% of the total variance can be explained by the model, indicating the model fits the data well. One can estimate the CPU time,

$$\begin{aligned} T_{CPU} &= -22.5 + 0.66 \times \frac{M(M-1)}{2} + 0.79L \\ &= -22.5 + 0.33M^2 - 0.33M + 0.79L, \end{aligned}$$

where M is the number of nodes and L time length in a thousand unit, i.e. $L = 30$ for 3×10^4 data points. Note that the computational time is measured in parallel computing of 20 workers in Matlab on 10 core Apple M2 Pro CPU with 16 GB RAM.

7.4 Summary

In this section, we introduced a novel method, APMSD, based on the randomisation of pc_1 percent of instantaneous amplitudes and pc_2 percent of instantaneous phases to generate surrogate data. This approach allows us to test connectivity among nodes using hypothesis tests and gain a better understanding of the source of connectivity in terms of both the amplitude and phase of a given signal. For some pairs of (pc_1, pc_2), it can also infer the network structure correctly in a wide range of applications, including deterministic discrete-time systems, deterministic or stochastic continuous-time systems. We evaluated the method in the sense of not only dynamics, but also network topology. We applied the method to the data generated based on various network types, including Watts-Strogatz small world, Erdös-Rényi random and Barabási-Albert scale free networks. For networks with 30 nodes and comparable density, the method consistently produced similar results, suggesting that the network type does not significantly influence network inference. However, for larger networks, the method incurs significant computational effort, as CPU time is proportional to the square of the network size. Regarding network topology, we found that network size and density affect the method's performance. Specifically, the method can capture connectivity more accurately for smaller and more sparse networks.

In conclusion, the APMSD method considers the combination of amplitude correlation and phase synchronisation as sources of connectivity. It provides information on their contributions to network connectivity and achieves perfect inference for certain pairs of randomisation parameters (pc_1, pc_2). However, applying this method to real-life data is challenging unless the initial network is known beforehand, as determining the randomisation parameters remains an open question.

Chapter 8

Conclusion & Outlook

8.1 Conclusion

In this thesis, we suggested a novel method, APMSD, to infer the network topology from synthetic data generated from various dynamics and network types. It also allows us to investigate the source of connectivity in a network in terms of combination of instantaneous amplitudes and phases.

In Chapter 2, we introduced the definitions and properties of bidirectional and weighted networks, along with well-known network models such as ring lattice, random, small-world, and scale-free networks. This foundation sets the stage for the discussion of network inference in the subsequent chapters.

In Chapter 3, we briefly discussed the non-linear dynamical system and chaos theory to elaborate the dynamical behaviours of models used in the subsequent chapters. In this sense, we discuss the numerical integration methods for ordinary and stochastic differential equations to solve the model equations. We used Lyapunov exponents to measure the average exponential divergence (or convergence) through the trajectory using the approach in [8].

Chapter 4 discusses the dynamical behaviour of the system in terms of synchronisation. Although chaotic synchronisation may seem paradoxical due to the sensitivity of chaotic systems to initial conditions and the idea of synchrony implying harmony among system units, it can occur under certain coupling configurations. We illustrate this phenomenon through examples of CS, PS, and IPS within a network context. We delve into the concepts of phases, amplitudes, and frequencies, as well as the analytic signal, using the HT as introduced by Gabor [23]. Furthermore, to assess phase and amplitude order within a network, we employ three measures from the literature. This comprehensive approach allows us to thoroughly investigate the synchronisation behaviour of the systems examined subsequently.

Chapter 5 is dedicated to discussing information-theoretic approaches with applications. In the context of network inference, we find that information-theoretic approaches outperform PC in capturing nonlinear and linear relations. Additionally, we explored various probes such as the sine of the signal, time derivative, etc., and concluded that the effectiveness of a probe depends on its ability to differentiate between connected and unconnected pairs. We discussed the effect of structural behaviour on network inference, such as random, small-world, scale-free networks. We did not observe any significant effect of network types with forty nodes and comparable density on network inference.

After some discussion on the dynamical behaviour of the systems in Chapters 3 and 4 and structural properties of a network in Chapter 5, we focused on the thresholding problem in the last two chapters of this thesis. Chapter 6 is dedicated to discussing network inference through statistical tests using RSD and TWSD, proposed by the author in [2]. While the former destroys all linear and nonlinear relationships in the data, the latter destroys the phase synchronisation but preserves the linear correlation by randomising the phases using the FT. However, there are some degree of amplitude correlation and phase synchronisation present among the nodes even if they are not connected because they are actually connected through some other node/s. When the surrogate data destroy all linear and / or non-linear relations, the method captures the extra links in many simulations that results in high FPR.

To address this issue, we presented a novel method in network inference: the method of statistical tests using APMSD. In this approach, we introduced two percentage parameters to randomise instantaneous amplitudes and phases, partially using the concept of analytic signal derived by HT. This approach allows us to infer the network structure successfully for some pairs of randomisation parameters (pc_1, pc_2), while also providing information about the source of connectivity as a combination of amplitude correlation and phase synchronisation.

8.2 Outlook

As a further investigation built on the APMSD method introduced in this thesis, it is possible to train a machine learning algorithm to establish a relationship between the dynamics of the system and the pairs of percentages, (pc_1, pc_2), leading to perfect inference because we found no straightforward relation between them. This approach allows our method to become applicable to time series data without prior knowledge of the initial network.

Furthermore, in this thesis, we explored network inference via hypothesis testing using surrogate data, which lacks information about the source of connectivity. Alternatively, one can address the same problem of network inference without thresholding by framing it as a classification problem in multidimensional space. Such a classification algorithm may consider various factors that influence network inference, such as amplitude correlation and phase synchronisation.

For example, we plotted pairs of pairwise phase order, as defined in Section 4.4.3, and the absolute value of Pearson correlation (PC) for both the HR system (see Fig. 8.1(a)) and the circle map (see Fig. 8.1(b)), using the same data discussed in Chapters 6 and 7. In this plot, connected and unconnected pairs are represented by blue and red circles, respectively. Panel (a) demonstrates a clear distinction between the groups, while panel (b) exhibits less pronounced separation. Consequently, it is feasible to investigate network inference as a classification problem, taking into account factors that influence network inference.

Another potential extension of network inference could involve considering causal relations using methods such as transfer entropy and/or Granger causality after inferring the

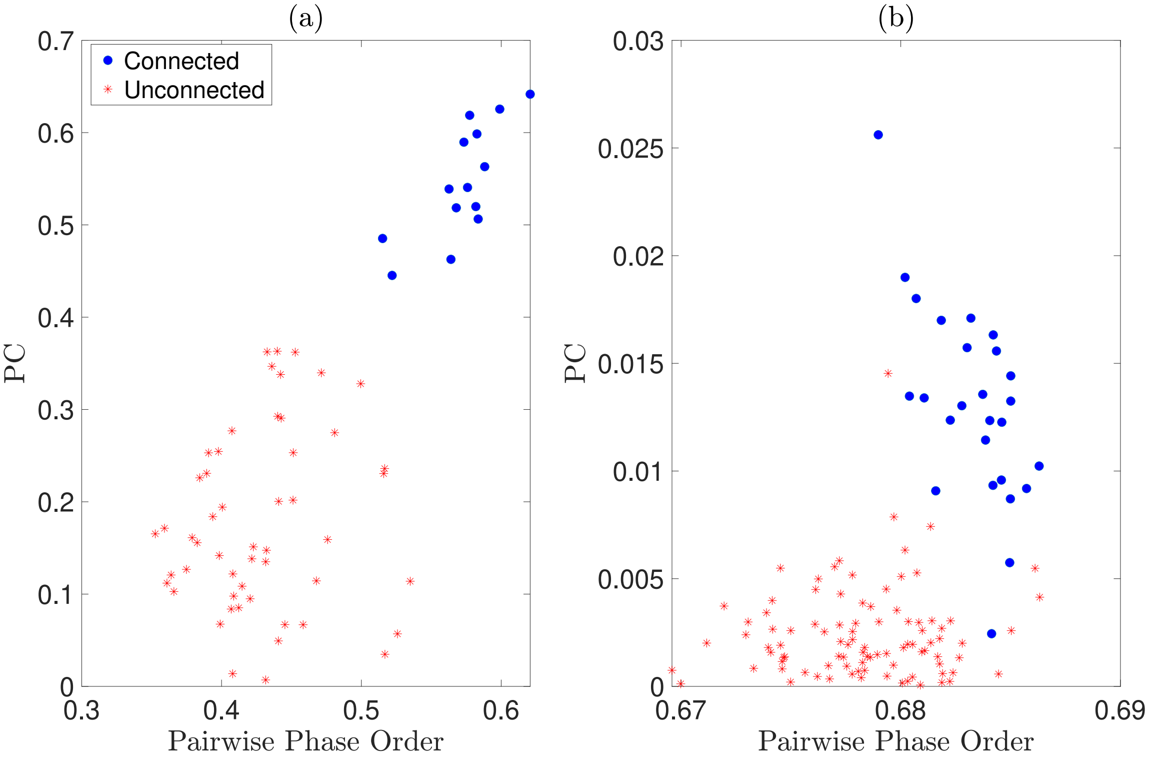

Figure 8.1: **Classification problem**. Network inference discussed in this thesis can be re-framed as a classification problem. Here, we demonstrate this concept using two factors influencing network inference: amplitude correlation and phase synchronisation. Plots of the pairs of pairwise phase order (defined in Sec. 4.4.3) and PC are shown for the trajectories from (a) HR system utilised to produce Fig. 6.6 and (b) circle map utilised to produce Fig. 6.3. Here, connected and unconnected pairs are indicated by blue and red circles, respectively.

skeleton of the network. Although this thesis focuses primarily on bidirectional network inference, causal interactions are also crucial in many scenarios. They can be investigated through the aforementioned methods after inferring the bidirectional network. However, this approach may incur significant computational costs.

Bibliography

[1] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47, 2002.

[2] Chris G Antonopoulos. Network inference combining mutual information rate and statistical tests. *Communications in Nonlinear Science and Numerical Simulation*, 116, 2023.

[3] Murilo S Baptista, Rero M Rubinger, Emilson R Viana, José C Sartorelli, Ulrich Parlitz, and Celso Grebogi. Mutual information rate and bounds for it. *PLoS One*, 7(10), 2012.

[4] Albert-Laszlo Barabasi. *Linked: How everything is connected to everything else and what it means for business, science and everyday life.* Basic Books, 2003.

[5] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[6] Elisa Benedetti, Maja Pučić-Baković, Toma Keser, Nathalie Gerstner, Mustafa Büyüközkan, Tamara Štambuk, Maurice HJ Selman, Igor Rudan, Ozren Polašek, Caroline Hayward, et al. A strategy to incorporate prior knowledge into correlation network cutoff selection. *Nature Communications*, 11(1):5153, 2020.

[7] Giancarlo Benettin, Luigi Galgani, Antonio Giorgilli, and Jean-Marie Strelcyn. Lyapunov characteristic exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them. part 1: Theory. *Meccanica*, 15(1):9–20, 1980.

[8] Giancarlo Benettin, Luigi Galgani, Antonio Giorgilli, and Jean-Marie Strelcyn. Lyapunov characteristic exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them. Part 2: Numerical application. *Meccanica*, 15(1):21–30, 1980.

[9] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995.

[10] Yoav Benjamini and Daniel Yekutieli. The control of the false discovery rate in multiple testing under dependency. *Annals of Statistics*, pages 1165–1188, 2001.

[11] Yoav Benjamini and Daniel Yekutieli. False discovery rate–adjusted multiple confidence intervals for selected parameters. *Journal of the American Statistical Association*, 100(469):71–81, 2005.

[12] E. Bianco-Martinez, N. Rubido, Ch. G. Antonopoulos, and M. S. Baptista. Successful network inference from time-series data using mutual information rate. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 26(4), 2016.

[13] Boualem Boashash. Estimating and interpreting the instantaneous frequency of a signal. i. fundamentals. *Proceedings of the IEEE*, 80(4):520–538, 1992.

[14] Boualem Boashash. Estimating and interpreting the instantaneous frequency of a signal. ii. algorithms and applications. *Proceedings of the IEEE*, 80(4):540–568, 1992.

[15] Stefano Boccaletti, Jürgen Kurths, Grigory Osipov, DL Valladares, and CS Zhou. The synchronization of chaotic systems. *Physics Reports*, 366(1-2):1–101, 2002.

[16] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4-5):175–308, 2006.

[17] Przemyslaw Bogacki and Lawrence F Shampine. A 3 (2) pair of Runge-Kutta formulas. *Applied Mathematics Letters*, 2(4):321–325, 1989.

[18] John Adrian Bondy, Uppaluri Siva Ramachandra Murty, et al. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[19] William E Boyce and Richard C DiPrima. *Elementary differential equations and boundary value problems*. John Wiley & Sons, Inc. All rights reserved., 2012.

[20] Georges A Darbellay and Igor Vajda. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, 45(4):1315–1321, 1999.

[21] John R Dormand and Peter J Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, 1980.

[22] Paul Erdős, Alfréd Rényi, et al. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[23] Dennis Gabor. Theory of communication. Part 1: The analysis of information. *Journal of the Institution of Electrical Engineers-part III: Radio and Communication Engineering*, 93(26):429–441, 1946.

[24] Carlos Gias. Phase randomization. `https://uk.mathworks.com/matlabcentral/fileexchange/32621-phase-randomization`. Accessed: 06-08-2024.

[25] James Gleick. *Chaos: Making a new science*. Penguin, 2008.

[26] Yong Kheng Goh, Haslifah M Hasim, and Chris G Antonopoulos. Inference of financial networks using the normalised mutual information rate. *PloS One*, 13(2):e0192160, 2018.

[27] Jesús Gómez-Gardeñes, Gorka Zamora-López, Yamir Moreno, and Alex Arenas. From modular to centralized organization of synchronization in functional areas of the cat cerebral cortex. *PLoS One*, 5(8):e12313, 2010.

[28] Arthur Gretton, Ralf Herbrich, Alexander Smola, Olivier Bousquet, Bernhard Schölkopf, et al. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6, 2005.

[29] Jordan Hart, Michael Nash Weiss, Daniel Franks, and Lauren Brent. Bison: A bayesian framework for inference of social networks. *Methods in Ecology and Evolution*, 14(9):2411–2420, 2023.

[30] Norden E Huang, Zheng Shen, Steven R Long, Manli C Wu, Hsing H Shih, Quanan Zheng, Nai-Chyuan Yen, Chi Chao Tung, and Henry H Liu. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1971):903–995, 1998.

[31] Norden E Huang, Zhaohua Wu, Steven R Long, Kenneth C Arnold, Xianyao Chen, and Karin Blank. On instantaneous frequency. *Advances in Adaptive Data Analysis*, 1(02):177–229, 2009.

[32] Vân Anh Huynh-Thu and Guido Sanguinetti. Gene regulatory network inference: an introductory survey. *Gene Regulatory Networks: Methods and Protocols*, pages 1–23, 2019.

[33] Luis Iglesias-Martinez, Barbara Kegel, and Walter Kolch. Kboost: A new method to infer gene regulatory networks from gene expression data. *Scientific Reports*, 11, 07 2021.

[34] T Kathleen, D Tim, and A James. *Chaos: an introduction to dynamical systems.* Springer, New York, NY, USA, 1996.

[35] Peter E Kloeden and Eckhard Platen. *Stochastic differential equations.* Springer, 1992.

[36] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical Review E*, 69(6), 2004.

[37] Yoshiki Kuramoto. International symposium on mathematical problems in theoretical physics. *Lecture Notes in Physics*, 30:420, 1975.

[38] Gemma Lancaster, Dmytro Iatsenko, Aleksandra Pidde, Valentina Ticcinelli, and Aneta Stefanovska. Surrogate data for hypothesis testing of physical systems. *Physics Reports*, 748:1–60, 2018.

[39] Ling-Hsiao Lyu. Appendix c. higher-order numerical integrations. `http://www.ss.ncu.edu.tw/~lyu/lecture_files_en/lyu_NSSP_Notes/Lyu_NSSP_AppendixC.pdf`, journal=Numerical Simulation of Space Plasmas (I) Lecture Notes, Aug 2016.

[40] Parul Maheshwari and Réka Albert. Network model and analysis of the spread of covid-19 with social distancing. *Applied Network Science*, 5(1):100, 2020.

[41] Robert M May. Simple mathematical models with very complicated dynamics. *Nature*, 261(5560):459–467, 1976.

[42] Cleve Moler. Ordinary differential equation solvers ode23 and ode45. `https://blogs.mathworks.com/cleve/2014/05/26/ordinary-differential-equation-solvers-ode23-and-ode45/`, 2014. Last accessed 9 January 2024.

[43] Young-Il Moon, Balaji Rajagopalan, and Upmanu Lall. Estimation of mutual information using kernel density estimators. *Physical Review E*, 52(3):2318, 1995.

[44] Mark Newman. *Networks*. Oxford University Press, 2018.

[45] Alan V Oppenheim and George C Verghese. *Signals, systems & inference*. Pearson London, 2017.

[46] Alan V. Oppenheim and Schafer Ronald W. *Discrete-Time Signal Processing*. Pearson New International Edition, 2014.

[47] Giuliano Andrea Pagani and Marco Aiello. The power grid as a complex network: a survey. *Physica A: Statistical Mechanics and its Applications*, 392(11):2688–2700, 2013.

[48] Roxana Pamfil, Nisara Sriwattanaworachai, Shaan Desai, Philip Pilgerstorfer, Konstantinos Georgatzis, Paul Beaumont, and Bryon Aragam. Dynotears: Structure learning from time-series data. In *International Conference on Artificial Intelligence and Statistics*, pages 1595–1605. PMLR, 2020.

[49] UCSC Physics. Ucsc physics demonstration room. `https://ucscphysicsdemo.sites.ucsc.edu/physics-5b6b-demos/oscillations-and-waves/pendulum/`, 2024. Last accessed 9 January 2024.

[50] Anna Pidnebesna, Pavel Sanda, Adam Kalina, Jiri Hammer, Petr Marusic, Kamil Vlcek, and Jaroslav Hlinka. Tackling the challenges of group network inference from intracranial eeg data. *Frontiers in Neuroscience*, 16, 2022.

[51] Arkady Pikovsky, Michael Rosenblum, Jürgen Kurths, and A Synchronization. A universal concept in nonlinear sciences. *Self*, 2:3, 2001.

[52] Márton Pósfai and Albert-László Barabási. *Network science*. Citeseer, 2016.

[53] Dean Prichard and James Theiler. Generating surrogate data for time series with several simultaneously measured variables. *Physical Review Letters*, 73(7):951, 1994.

[54] Jagath C Rajapakse, Yang Wang, Xuebin Zheng, and Juan Zhou. Probabilistic framework for brain connectivity from functional mr images. *IEEE Transactions on Medical Imaging*, 27(6):825–833, 2008.

[55] Michael G Rosenblum, Arkady S Pikovsky, and Jürgen Kurths. Phase synchronization of chaotic oscillators. *Physical Review Letters*, 76(11):1804, 1996.

[56] Sigrid Rouam. *False Discovery Rate (FDR)*, pages 731–732. Springer New York, 2013.

[57] Nicolás Rubido, Arturo C Martí, Ezequiel Bianco-Martínez, Celso Grebogi, Murilo S Baptista, and Cristina Masoller. Exact detection of direct links in networks of interacting dynamical units. *New Journal of Physics*, 16(9):093010, 2014.

[58] Elizabeth L Sander, J Timothy Wootton, and Stefano Allesina. Ecological network inference from long-term presence-absence data. *Scientific Reports*, 7(1):7154, 2017.

[59] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[60] Ralf Steuer, Jürgen Kurths, Carsten O Daub, Janko Weise, and Joachim Selbig. The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, 2002.

[61] Steven H Strogatz. From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D: Nonlinear Phenomena*, 143(1-4):1–20, 2000.

[62] Changcheng Sun, Fei Yang, Chunfang Wang, Zhonghan Wang, Ying Zhang, Dong Ming, and Jingang Du. Mutual information-based brain network analysis in post-stroke patients with different levels of depression. *Frontiers in Human Neuroscience*, 12:285, 2018.

[63] Stefan Thurner, Rudolf Hanel, and Peter Klimek. *Introduction to the theory of complex systems*. Oxford University Press, 2018.

[64] Giulio Tirabassi, Ricardo Sevilla-Escoboza, Javier M Buldú, and Cristina Masoller. Inferring the connectivity of coupled oscillators from time-series statistical similarity analysis. *Scientific Reports*, 5(1):10829, 2015.

[65] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’networks. *Nature*, 393(6684):440–442, 1998.