Inferring oscillatory dynamics from data

Rok Cestnik

This dissertation has been accomplished within the COSMOS project. The project has recieved funding from the European Union's Horizon 2020 research innovation programme under the Marie Skłodowska-Curie grant agreement No. 642563.

This PhD thesis was embedded within Amsterdam Movement Sciences research institute, at the Department of Human Movement Sciences, Vrije Universiteit Amsterdam, the Netherlands.

ISBN: 978-94-93184-46-6
Printed by: Proefschrift-AIO

VRIJE UNIVERSITEIT

Inferring oscillatory dynamics from data

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad Doctor
aan de Vrije Universiteit Amsterdam
en Universität Potsdam,
op gezag van de rectores magnifici
prof.dr. V. Subramaniam
en prof.dr. O. Günther,
in het openbaar te verdedigen
ten overstaan van de promotiecommissie
van de Faculteit der Gedrags- en Bewegingswetenschappen
op donderdag 2 juli 2020 om 11.45 uur
in een online bijeenkomst van de universiteit,
De Boelelaan 1105

door

Rok Cestnik

geboren te Ljubljana, Slovenië

promotoren: prof.dr. M. Rosenblum
prof.dr. A. Daffertshofer

Inferring oscillatory dynamics from data

Rok Cestnik

Thesis submitted for a
joint PhD degree
undertaken at

&

Institute for
Physics and Astronomy
University Potsdam
Potsdam, Germany

Department of
Human Movement Sciences
Vrije Universiteit Amsterdam
Amsterdam, Netherlands

2016-2020

There is nothing more extraordinary than reality

Mary Ellen Mark

Contents

Chapter 1

Introduction

Oscillators are systems with a swinging output. They can be purely periodic, meaning that their state repeats after a certain period, or less regular due to internal or external factors. Within dynamical systems they emerge from different mathematical frameworks, the typical being differential equations and dynamical maps. We will focus only on continuous oscillators, therefore we will not consider maps.

Both ordinary differential equations (ODEs) as well as delay differential equations (DDEs) are often used to describe oscillatory systems. ODEs describe the relationship between the variables and their derivatives (typically in time), and DDEs can also include delayed values of those variables. While ODEs cover finite dimensional systems, systems with delay are in principle infinitely dimensional [1]. Because of that, DDEs are typically harder to solve, but they offer rich dynamics. Some trajectories of oscillatory systems are depicted in Figure 1.1. They correspond to the Mackey-Glass equation [2], a DDE that we will refer to later in the Chapter 4. The orbits are depicted in a time-delay embedded space, since the example system only yields a single time series, $x(t)$.

1.1 Oscillator classification

One can classify oscillators by their dynamical behaviour. Here I will briefly introduce three different types of oscillators: self-sustained, excitable and stochastic.

1.1.1 Self-sustained oscillators

Self-sustained oscillators oscillate indefinitely without any stimulation. Physically, this means they incorporate their own power source. Some examples of self-sustained oscillators include the pendulum clock, flashing fireflies, beating hearts, etc.

Non-chaotic self-sustained oscillators are dynamical systems with an attractive limit cycle. If such a system is left unperturbed, it converges to this limit cycle, where it then stays oscillating periodically with its natural frequency.

Chaotic self-sustained oscillators are characterized by a positive Lyapunov exponent λ [3]. This exponent quantifies the rate of divergence of nearby trajectories. At any point in the state space $\vec{x}_0$ the short-term time evolution of the distance between two nearby trajectories $\vec{\delta}$ grows exponentially:

$$\left| [\vec{x}_0]_t - [\vec{x}_0 + \vec{\delta}]_t \right| = |\vec{\delta}| e^{\lambda t} \tag{1.1}$$

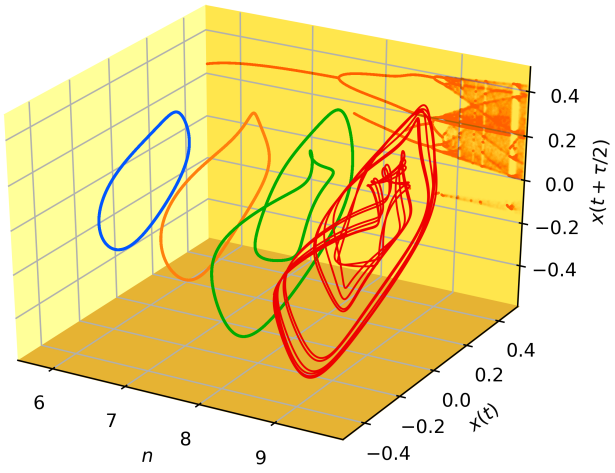


FIGURE 1.1: Time embedded trajectories of the Mackey-Glass equation (4.7). Four different trajectories corresponding to four different parameters $n = 6, 7, 8$ and 9 depicted in blue orange green and red respectively. In the background the bifurcation diagram is shown in orange.

here I denote the operator of time evolution with $[\cdot]_t$. The local Lyapunov exponent at the point $\vec{x}_0$ in direction $\vec{\delta}$ and can be quantified as:

$$\lambda_{\vec{\delta}} = \lim_{t\to 0}\lim_{\delta\to 0}\frac{1}{t}\left(\log\left|[\vec{x}_0]_t - [\vec{x}_0+\vec{\delta}]_t\right| - \log|\vec{\delta}|\right) \tag{1.2}$$

The vector $\vec{\delta}$ specifies the direction of the difference between two trajectories and it should be small in absolute value because exponential scaling (1.1) only holds for nearby trajectories, that is why a limit of $\delta \to 0$ is taken. The limit in time $t \to 0$ specifies that this is a local exponent. Lyapunov exponents can also refer to global exponents that describe the divergence over longer times and therefore do not depend on a single state point $\vec{x}_0$. They can be expressed mathematically with a similar expression as (1.2), just with time going to infinity:

$$\lambda = \lim_{t\to\infty}\lim_{\delta\to 0}\frac{1}{t}\left(\ldots\right) \tag{1.3}$$

In practice, global Lyapunov exponents (1.3) can be estimated by averaging the local Lyapunov exponents (1.2) over the attractor. A system has as many global Lyapunov exponents as its dimensionality and if at least one is positive, then the system is chaotic. An example of a chaotic oscillator is the Roessler system [4]:

$$\begin{aligned}\dot{x} &= -y - z\\ \dot{y} &= x + ay\\ \dot{z} &= b + z(x - c)\end{aligned} \tag{1.4}$$

with $a = b = 0.2$ and $c = 5.7$, see Figure 1.2 for a trajectory realization. This example is three dimensional, meaning it has three Lyapunov exponents. Only one is positive, and that suffices for chaos. The second Lyapunov exponent is zero, which corresponds to the direction of the flow. And the third Lyapunov exponent is negative, accounting for the attractor being attractive, i.e. the sum of all Lyapunov exponents of any attractor is negative.

As a comparison, the maximal Lyapunov exponent of before-mentioned limit cycle oscillators is always zero. On any limit cycle the direction $\vec{\delta}$ corresponding to the fastest divergence (and therefore the largest local Lyapunov exponent) is tangential to the limit cycle. Hence the flow on the limit cycle is marginally stable, i.e. two trajectories starting anywhere on the limit cycle over time neither diverge nor converge, their separation (averaged over 1 period) remains constant.

1.1.2 Excitable oscillators

Excitable systems are those characterized by having a resting state and an excited state. If left unperturbed, their autonomous dynamics will take them to their resting state, which is typically a stable fixed point. When stimulated with an external signal, they can reach their excited state. From there they take a different path to return back to the resting state. A continuously stimulated excitable system might spend very little time in its resting state and can thus appear to be oscillating relatively regularly [5].

The time it takes for the unperturbed system to return from the excited state back to the resting state is called excursion time t_e. It is a natural and intuitive quantity, but can be difficult to define non-arbitrarily because it is difficult to define exact boundaries of the resting state, see Figure 1.3.

An example of an excitable system defined with ordinary differential equations is the FitzHugh-Nagumo neuronal model [6]:

$$\begin{aligned} \dot{x} &= x - x^3/3 - y \\ \dot{y} &= (x + 1.05)/10 \end{aligned} \tag{1.5}$$

Its resting state is a stable fixed point. I depict the time it takes to reach the $\epsilon = 0.001$ vicinity of the fixed point from any point in the state space in color in Figure 1.3. In the figure the boundary that determines the excited state can be clearly seen as a sharp discontinuity in the time to reach the resting state (red-green boundary).

1.1.3 Stochastic oscillators

Stochastic oscillators fall into a separate category: every oscillator can be stochastic when combined with noise. The noise can be considered part of the system, e.g. an excitable oscillator perturbed by a stochastic force can be considered as one system, and that union makes a stochastic oscillator. Stochastic oscillators are not deterministic which means that their evolution cannot be exactly determined. That is reminiscent of chaotic systems, where the positive divergence makes it hard to accurately predict future states, but in the case of stochastic systems it is not difficult, it is fundamentally impossible.

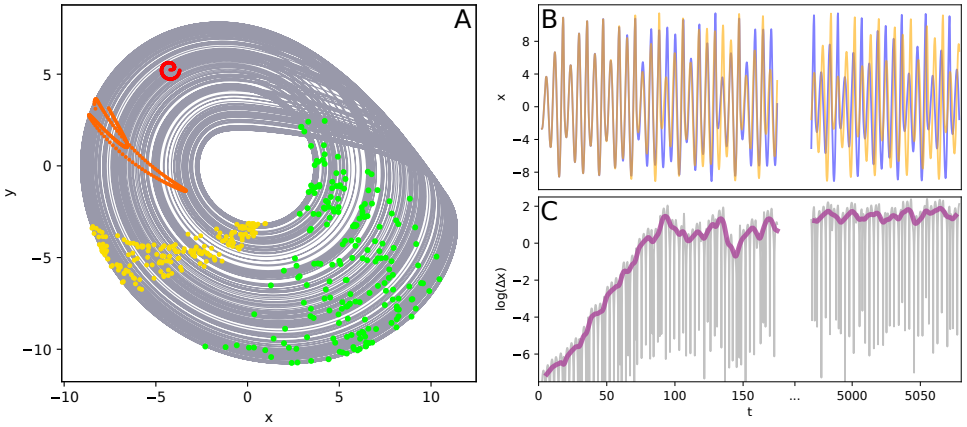


FIGURE 1.2: A: chaotic attractor of the Roessler system [4] (1.4). A close group of starting points in red, time evolved for 3 rotations in orange, for 10 rotations in yellow and 100 rotations in green - demonstrating the diffusion within the system. B: two trajectories starting from a nearby point ($\delta = 0.001$) depicted with blue and orange. C: the logarithm of the difference between the two trajectories depicted in B. The time averaged difference is depicted with thick purple.

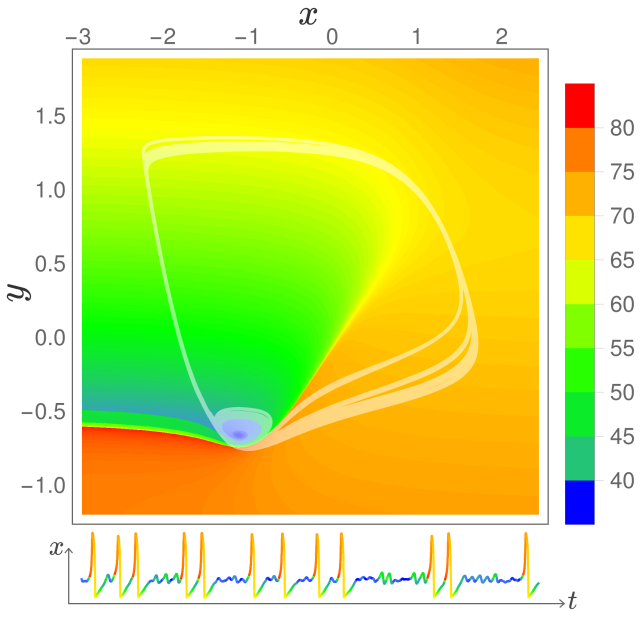


FIGURE 1.3: The state space of an excitable oscillator, FitzHugh-Nagumo model [6] (1.5). The time it takes to reach the $\delta = 0.01$ vicinity of the attractive fixed point is depicted in color. A trajectory of the system perturbed by correlated noise is shown in transparent white and the x component in time below, the colors match the ones above.

1.2 Phase

Dynamics of a system on a stable limit cycle are essentially one-dimensional and may be described by a single variable, called the phase. By fulfilling the condition of uniform growth under autonomous dynamics: $\dot{\varphi} = \omega$, the phase is uniquely defined up to a constant offset. For autonomous dynamics it is therefore congruent with time.

Limit cycles are periodic, meaning that their state repeats after a certain amount of time we call the period T_0. It is therefore common to consider the phase wrapped on a finite interval that represents one period. The size of this interval is arbitrary, but commonly it is either $(0, 1)$ or $(0, 2\pi)$. I will be using $(0, 2\pi)$ in this thesis. Importantly, the concept of phase can be extended to states beyond the limit cycle by using the notion of asymptotic phase. The idea is that any state that is within the limit cycle's basin of attraction will asymptotically approach it and therefore one can assign a phase to any state by comparing it to a reference state on the limit cycle. The surfaces (curves in 2D) in the state space that correspond to the same asymptotic phase are called isochrons [7], see Figure 1.4 for examples. The notion of asymptotic phase cannot be applied as rigorously to chaotic oscillators as it can be to limit cycle oscillators. This is because of the lack of a reference point. On a chaotic attractor an arbitrarily condensed group of points will in time diffuse evenly over the entire attractor (see Figure 1.2A). Nevertheless, one can make an attempt at "roughly" describing the dynamics in terms of something similar to the phase [8]. The same holds true for stochastic oscillators. They cannot be rigorously assigned asymptotic phase, but can be approximately described with a quantity similar to phase [9].

Excitable oscillators can also be assigned a notion analogous to the phase - a one-dimensional variable that defines the state and follows an equation of a particular form. While for limit cycle oscillators that equation is uniform growth: $\dot{\varphi} = \omega$, for excitable systems that is the quadratic integrate and fire equation: $\dot{\varphi} = \varphi(\varphi - c)$ [10], where the constant c defines the transition point between the resting and excited state and the phase resets after reaching a certain threshold since the equation otherwise has no upper boundary. From here on, I will use the term *phase* to refer to the asymptotic phase of a limit cycle oscillator only.

1.2.1 Numerical phase determination

Numerically, there is a simple scheme to determine the asymptotic phase of any point within the basin of attraction. The phase is periodic, meaning that after a time evolution of exactly one period it remains unchanged:

$$\varphi\big([\vec{x}_0]_t\big) = \varphi\big([\vec{x}_0]_{t+nT_0}\big), \qquad \forall n \in \mathbb{Z}$$

Therefore, given a point $\vec{x_0}$, one simply has to let the system evolve in time for several natural periods T_0 until the system is close enough to its limit cycle. It is worth noting that in order to do this reliably, the natural period has to be determined with a high precision, otherwise a drift effect may build up. In practice, this can be accomplished with proper integration schemes, such as the so-called Hénon trick [11].

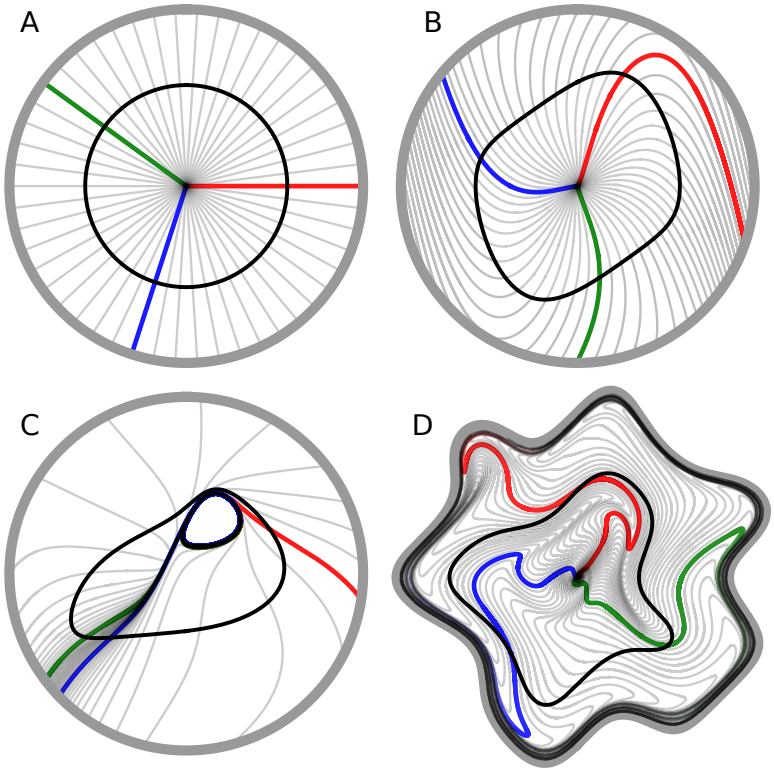


FIGURE 1.4: The isochronal structure of 4 oscillators. The limit cycles are depicted with a black line, while the isochrons are depicted with gray. Three of the isochrons are highlighted with colors: red corresponding to phase 0, green to $\frac{4}{5}\pi$ and blue to $\frac{7}{5}\pi$. There are several things to notice in this figure:

- the oscillators can rotate in time either clockwise (B,D) or counter-clockwise (A,C), depending on the combination of their dynamics and spatial representation. Notice the three colored isochrons, they represent the same phase in all oscillators, 0 (red), $\frac{4}{5}\pi$ (green) and $\frac{7}{5}\pi$ (blue). Indeed this is trivial, they are all two dimensional oscillators and if we switch the x and y axis the resulting spatial representation depicts the oscillator rotation in the other direction.

- the curvature of isochrons. Oscillator A is an ideal case with uniform radial isochrons, in general the isochrons are curved as the other three examples indicate.

- the density of isochrons. Oscillator A and B both still have rather uniform isochrons, but we see in oscillator C that the density can vary dramatically, which translates to varying rotation speed in time. Notice again the three colored isochrons, green and blue are very close together even though they represent a substantial phase difference.

- the basin of attraction. Examples A and B have limit cycles whose basins of attraction span the entire xy plane. With oscillator C we can notice a rounded empty region inside the limit cycle towards its top. That region is not void of isochrons by coincidence, it is not part of the basin of attraction of the limit cycle and therefore isochrons are not defined there. Another example of a non-trivial basin of attraction is oscillator D, which encompasses all of the features mentioned so far. It has a basin of attraction that is limited from the outside. Its frame, which for other oscillator is simply a circle (their basin of attraction spans towards infinity) is in its case the boundary of its basin of attraction, which is an unstable periodic orbit.

The isochrons were calculated with a background integration algorithm, which is publicly available, see *Isochrons* in the Software section at the end.

1.2.2 Phase from signal observations

In this section I present an approach for numerically extracting the phase from signal observations without the knowledge of dynamical equations (for determining phase with equations see Section 1.2.1). Some parts apply or can be generalized to higher dimensional systems, but for the most part I will consider a two dimensional limit cycle oscillator.

Protophase

The protophase θ was conceptualized as a variable that bijectively maps onto the true phase [12, 13]:

$$\begin{aligned} \theta &= f(\varphi) \\ \varphi &= f^{-1}(\theta) \end{aligned} \tag{1.6}$$

but as we will see later, this relation may in general be very difficult to fulfill because of non-trivial isochronal structure. Hence, it might be only approximately true. Therefore, I will define the protophase as any variable that is monotonically growing for the unperturbed system, but the growth need not be uniform:

$$\dot{\theta} > 0$$

If the limit cycle is known and it encloses the origin, then a paradigmatic example for the protophase is the polar angle:

$$\theta(x, y) = \pi + 2 \arctan\left(\frac{y}{x + \sqrt{x^2 + y^2}}\right) \tag{1.7}$$

If only one signal is measured, it can be embedded by generating another variable (or more), see later Section 1.3. This may come at the cost of the estimated phase accuracy. If a non-local embedding is used, the information of the phase is blurred, and even if the embedding is local, the limit cycle might fold in the embedded space which can make it difficult to find a protophase candidate. In such cases one typically uses case specific *ad hoc* methods, cf. [14].

Phase approximation

Once one has obtained an appropriate protophase, one can use it to estimate the phase. As a zero-order approximation let us assume radial isochrons, in which case the phase is obtained with the following transformation [15]:

$$\varphi(\theta) = \omega \int_0^\theta \frac{1}{\dot{\theta}_0(\theta)} \, d\theta \tag{1.8}$$

where $\dot{\theta}_0(\theta)$ is the velocity of the protophase on the limit cycle. Ideally, $\dot{\theta}_0$ is evaluated from a measurement of the unperturbed limit cycle, but if that is inaccessible it can be estimated from the trajectory average. If one measures only the protophase, then the zero approximation of the phase is the best approximation available. In the case of a perturbed system the zero approximation is not adequate for phase derivative

estimations due to the error of neglecting the isochrons, see Section 6.1.1 for a more detailed argument.

1.2.3 Phase dynamics

Since there is a way of parameterizing limit cycle systems with only one variable, it is useful to derive a dynamical equation for that variable and determine the domain on which it is valid. This can be accomplished by expanding the dynamics as a series. On the limit cycle, as a direct consequence of the phase definition, the autonomous oscillatory state evolution can be described with the equation:

$$\dot{\varphi} = \omega \tag{1.9}$$

where ω is the natural angular velocity of the oscillator. For small amplitudes the linear approximation of the effect of scalar perturbations $p(t)$ acting on the oscillator can be described with the Winfree phase equation [16]:

$$\dot{\varphi} = \omega + Z(\varphi)p(t) + \mathcal{O}(\epsilon^2) \tag{1.10}$$

where ϵ is the strength of the perturbation $p(t)$ and $Z(\varphi)$ is the so-called phase response curve, which quantifies the phase susceptibility to be changed by stimulation. To increase the accuracy of the phase description beyond the second order, amplitude dynamics would have to be considered as well [17]. Since amplitude is asymptotically stable, with a negative Lyapunov exponent, the linear amplitude equation may be written as follows:

$$\dot{A} = \mu A + Y(\varphi)p(t) + \mathcal{O}(\epsilon^2) \tag{1.11}$$

where the parameter $\mu < 0$ denotes the Lyapunov exponent and $Y(\varphi)$ the amplitude response curve. All further terms in the phase expansion (1.10) would depend on both phase and amplitude. I will not consider higher-order expansions in this thesis.

What phase dynamics cannot describe?

Phase dynamics is excellent for describing self-sustained oscillators in the vicinity of their limit cycles. However, it is clearly a simplification and as such does not capture all the features of the original system. If only the phase is considered, then all amplitude effects and higher-dimensional effects will not be present. These include the dependence of the phase response on amplitude which, for example, may yield shear-induced folding [18, 19]. Further, no matter the input, a self-sustained phase oscillator cannot describe an excitable system while vice-versa is possible: suppose a quadratic integrate and fire oscillator is fed a constant positive input I, $\dot{\varphi} = \varphi(\varphi - c) + I$. Then for $I > c^2/4$ the system will oscillate by itself with a constant frequency and can be parameterized as a self-sustained phase oscillator. A practical example of this descriptive limitation is a neuron that receives a steady background input from its environment such that is fires almost periodically and can therefore be well described as a self-sustained limit cycle oscillator, however, if that same neuron received less input it would likely follow excitable dynamics [20, 21].

1.3 Signal embedding

Any system capable of reproducing oscillation is at least two dimensional, i.e. it takes at least two first-order ODEs to describe its dynamics. If only one variable is observed, in order to embed the trajectory into a higher-dimensional space, one has to generate more variables. There are several ways to do that [22, 23]. Here I briefly describe some schemes to embed a one dimensional signal into a two dimensional space. A natural classification starts with dividing the embedding schemes into local and non-local ones. Local schemes determine the value of the embedded variable only from the time localized signal, while non-local ones have no such restriction. In the following, I denote the observed signal with x and the embedded one with $\hat{x}$.

1.3.1 Local

Local embedding techniques determine the embedded variable at any time t from solely the time localized information of the signal $x(t)$. An example of such an embedding is the derivative embedding:

$$\hat{x}(t) = \frac{dx(t)}{dt}$$

The locality is a strong restriction. In fact, the complete family of local embeddings can be expressed as a power series of derivatives:

$$\hat{x}(t) = \sum_i \sum_j a_{ij} \Big(\frac{d^i x(t)}{dt^i} \Big)^j$$

In practice, higher derivatives are rarely used since each derivation amplifies any random fluctuations. Local embeddings represent, in general, a non-bijective (surjective) transformation from the true state variables to the embedded ones: $(x, y) \to (x, \hat{x})$. This means that structures (trajectories, isochrons, etc.) can fold and cross in the embedded space even if they do not cross in the true state space. Nevertheless, unlike in non-local embeddings, locality ensures that structures like isochrons do not scatter, see Figure 1.5. In practice, minor scattering can still take place due to strong forcing in combination with numerical discretization.

1.3.2 Non-local

Non-local embeddings have no restriction on locality and can therefore be in general expressed as a functional of the variable x at different times:

$$\hat{x}(t) = f[x\big(t \in (-\infty, \infty)\big)]$$

Non-local embeddings do not preserve isochrons. A point in space of a non-local embedding cannot be associated to any one asymptotic phase. An example of a non-local embedding is the Hilbert embedding:

$$\hat{x}(t) = H[x](t)$$

where H denotes the Hilbert transform [24], which can be elegantly expressed with the more commonly know Fourier transform F:

$$H[s](t) = F^{-1}\Big[-\mathbf{i}\ \mathrm{sgn}(\omega)F[s](\omega)\Big](t) \tag{1.12}$$

the bold "i" here represents the imaginary unit: $\mathbf{i} = \sqrt{-1}$. From Eq. (1.12) above one can realize that Hilbert transform decomposes the signal into its frequency components and then phase shifts each component by $\pi/2$. As a fun fact I mention that because a phase shift cannot be heard by human ears the Hilbert transform of an audio signal sounds the same as the original signal, even though it can look very different. Another example of a non-local embedding is the delay embedding [25]:

$$\hat{x}(t) = x(t-\tau)$$

It is commonly used, since it requires no additional computation. Figure 1.5 shows the mentioned embeddings side by side. A perturbed self-sustained oscillator is simulated to obtain the signal:

$$\begin{aligned} \dot{x} &= \quad y - \sin(y)x/2 \\ \dot{y} &= -x + \cos(x)y/2 + p(t) \end{aligned} \tag{1.13}$$

where the forcing $p(t)$ is correlated Gaussian noise, generated by an Ornstein-Uhlenbeck process [26]. Then, the x signal is embedded. The three different embeddings, along with the "true" state space are depicted side by side. Overlaid on the signal are the points corresponding to equal intervals of the asymptotic phase (red points), see Section 1.2.1 on how to determine phase. We can see that the contours of the phase remain relatively narrow when the local embedding is used, although they might still cross due to a non-bijective transformation $(x, y) \to (x, \hat{x})$. For the two non-local embeddings the phase contours are blurred due to the non-locality.

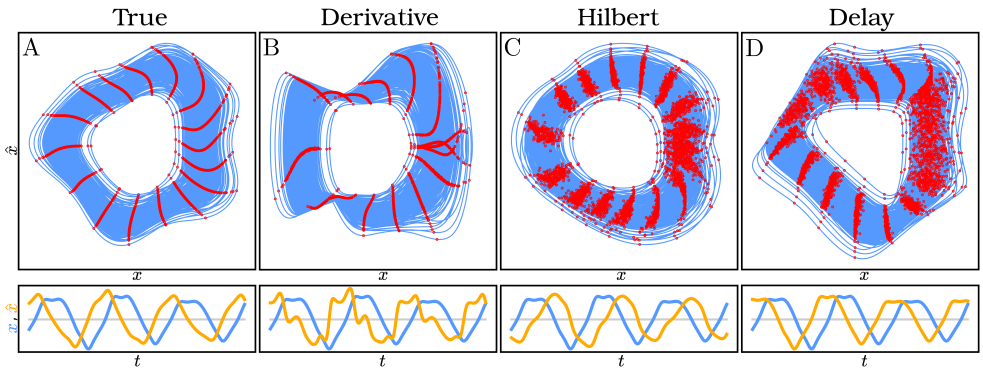


FIGURE 1.5: Different signal embedding methods. In blue, a perturbed trajectory of system (1.13) is depicted in its true state space (A) as well as three other embedding spaces (B,C,D). Below the state space representations are short depictions of the true x signal in blue and the embedded signal in orange. Red points overlaying the trajectories mark particular values of the asymptotic phase, effectively reflecting the isochronal structure. B: the derivative embedding is a local embedding and therefore does not scatter the isochronal structure, albeit the isochrons can cross due to non-bijectivity of the embedding transformation. C,D: The Hilbert embedding [24] and delay embedding [25] are both non-local which causes the isochronal structure to scatter.

1.4 Networks of oscillators

Oscillators can be coupled with other oscillators, yielding oscillatory networks which tend to be high-dimensional and are often more complex than the individual oscillatory nodes. In certain parameter regimes their dynamics can be simplified with phase reduction techniques [27], which often allow further simplification in certain limits, such as the Ott-Antonsen ansatz [28].

Networks of oscillators exhibit many interesting phenomena, the most notable being synchronization [15]. Synchronization is a phenomenon where the individual units come to have the same state due to interaction. It is observed in many natural systems, e.g. groups of fireflies synchronize their flashing cycle [29], pedestrians on a bridge synchronize their walking stride [30, 31], etc. Synchronization can be observed in networks of simple phase oscillators [32] as well as more complex ones, such as chaotic oscillators [15].

1.4.1 A seminal synchronization example

The Kuramoto model [32]:

$$\dot{\varphi}_i = \omega_i + \frac{\epsilon}{n} \sum_{j}^{n} \sin(\varphi_j - \varphi_i) \tag{1.14}$$

assumes oscillators can be well represented with their phases φ_i. Each of the n oscillators can have a different natural frequency ω_i and ϵ represents the overall strength of coupling. The oscillators are coupled symmetrically all-to-all. This simple model exhibits synchronization. To quantify this one can define an order parameter R as:

$$Re^{\mathbf{i}\psi} = \frac{1}{n} \sum_{i} e^{\mathbf{i}\varphi_i} \tag{1.15}$$

ψ here represents the mean phase, and R quantifies the degree of synchronization. A system that is fully synchronized has R equal to 1 and a fully asynchronous system has R equal to 0. In the thermodynamic limit $n \to \infty$, R undergoes a second-order phase transition, when the strength of coupling ϵ passes a positive critical value ϵ_c. Above ϵ_c, the system is synchronized. When the number of oscillators n is finite there are finite-size effects present and the transition is not "pure".

1.5 Inference

Inference is the process of deducing certain properties from data. Statistical inference requires some assumptions to be fulfilled. The set of assumptions can be called the "statistical model". More specific models with a greater number of assumptions require less data to infer. For example, when estimating the dependence of a variable y with respect to another variable x, if one assumes y is constant, then one only needs to make one measurement. On the other hand, if no assumption is made, then one has to observe many points to estimate the dependence.

1.5.1 Inferring dynamics

When observing a dynamical variable, one may want to infer laws that govern its dynamics [33]. Here a simple example is given. Let the measured variable be x. First one assumes a model, for example, that the dynamics obeys a one-dimensional first-order stochastic ODE:

$$\dot{x} = f(x) + \xi(t)$$

where $\xi(t)$ represents some noise term. Then, in order to infer dynamics, we have to determine f. One way to do that is to numerically evaluate the left-hand side derivative (assuming x is differentiable) and record the points $(\frac{\Delta x}{\Delta t}, x)$ which represent the function f. Due to the noise term $\xi(t)$ the points will be scattered around some region, but if enough points are measured a function can be fitted onto the points to obtain an approximation of f.

1.5.2 Inferring network connectivity

When inferring connectivity one first has to clarify what is meant by connectivity. That is to say, which criterion does one use to determine whether two nodes are connected or not. Let us first introduce some concepts:

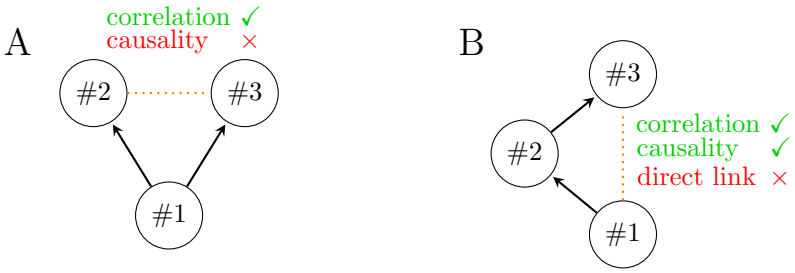
- **Correlation** is a mutual relationship between two variables. If variable #1 is correlated with variable #2, then variable #2 is necessarily correlated with variable #1.
- **Causality** is a concept describing the relationship between cause and effect. It is a stricter concept than correlation as it is typically not symmetric. If variable #1 is causally influencing variable #2, there need not be a causal influence from #2 to #1. Causal relationships almost always yield correlation, i.e. if variable #1 is causally influencing variable #2, they are likely correlated. The other way around is not true, if two variables are correlated it does not necessarily mean they causally interact - correlation does not imply causation.
- **Full model** of the interaction between two nodes is the quantification of all the processes that govern that interaction. It is a stricter concept than causality since it specifies the particularities of the causal interaction. For example, if a variable is influencing one variable via an integrator and another variable via a differentiator, it is causally influencing both, but the effect it has on them is very much different.

Correlations are straightforward to evaluate, but they are mere consequences of the underlying causal interactions. The correlations cannot be interpreted directly as indicators of causality. A paradigmatic example scheme illustrating this point is a triplet of oscillators where oscillator #1 is strongly driving oscillators #2 and #3, see Scheme 1.1A. Oscillators #2 and #3 are not coupled. There is a causal relationship between #1 - #2 and #1 - #3 and this is also reflected in their signals being correlated. However, signals #2 and #3 will also be correlated due to their common forcing, despite not being coupled and therefore not having any causal relationship.

In a similar way we can compare causality with the full model. It is certainly easier to infer causality, however, if one would like to simulate the system on a computer

there are many more details needed to determine the full model. An example scheme that illustrates this point is again of a triplet of oscillators where #1 is driving #2 and #2 is driving #3, see Scheme 1.1B. Oscillators #1 and #3 are not coupled directly. We can conclude that there is a causal relationship between oscillators #1 and #3, but it is only a consequence of the true connectivity (#1 and #3 are indirectly coupled via #2). To correctly infer the full model in this case would mean to identify that the connection between #1 and #3 is not present.

All of the three concepts are sometimes used to represent connectivity. Correlation is the easiest to determine and is also the most general, not requiring any assumptions. Causality is more difficult to determine. Causality inference schemes typically consider all the possible causal schemes and eliminate the ones that are not feasible with assumptions and information from observations [34, 35, 36, 37]. Arguably the hardest to infer is the full model of interactions. It requires strong assumption of the dynamics, and the inferred connectivity only has meaning within the scope of those dynamics. The method I present in the third chapter of this thesis is a full model inference.



Scheme 1.1: Two example schemes illustrating concepts of correlation, causality and the full model. A: unit #1 is directly influencing units #2 and #3. There is no interaction between units #2 and #3 but they are correlated due to common forcing. B: unit #1 is influencing unit #2 and unit #2 is influencing unit #3. In this case unit #1 is causally influencing unit #3 via an indirect connection through unit #2, however there is no direct link between them.

1.6 Artificial Neural Networks

Artificial neural networks are inspired by the naturally evolved biological neural networks - brains. They take certain principles from their operation, but they do not try to simulate the brain dynamics *per se*. The ideas already started in the middle of the 20th century with studies such as [38, 39], and caught on in the 1980s after the introduction of the backpropagation algorithm [40] which made practical implementations much more efficient. In the recent decade artificial neural networks again gained momentum due to the increase in computing power (Moore's Law [41] states that the computing power should roughly double every two years), and new concepts became practically feasible, such as deep learning [42, 43] which had a winning streak of achievements, some notable examples include the superiority of machine over man in the game of Go [44], a staggeringly swift improvement in image recognition [45], speech detection and speech generation [46] and many more. It is not the topic of this thesis to discuss the variety of neural networks in detail, but just to introduce them as a very easily adjustable universal model.

1.6.1 Feed-forward neural networks

Feed-forward networks are among the simplest artificial neural networks and were one of the first to be introduced [47]. They have connections that never form a cycle, therefore only allowing for information to flow in one direction, forward. Figure 1.6A depicts a feed-forward neural network.

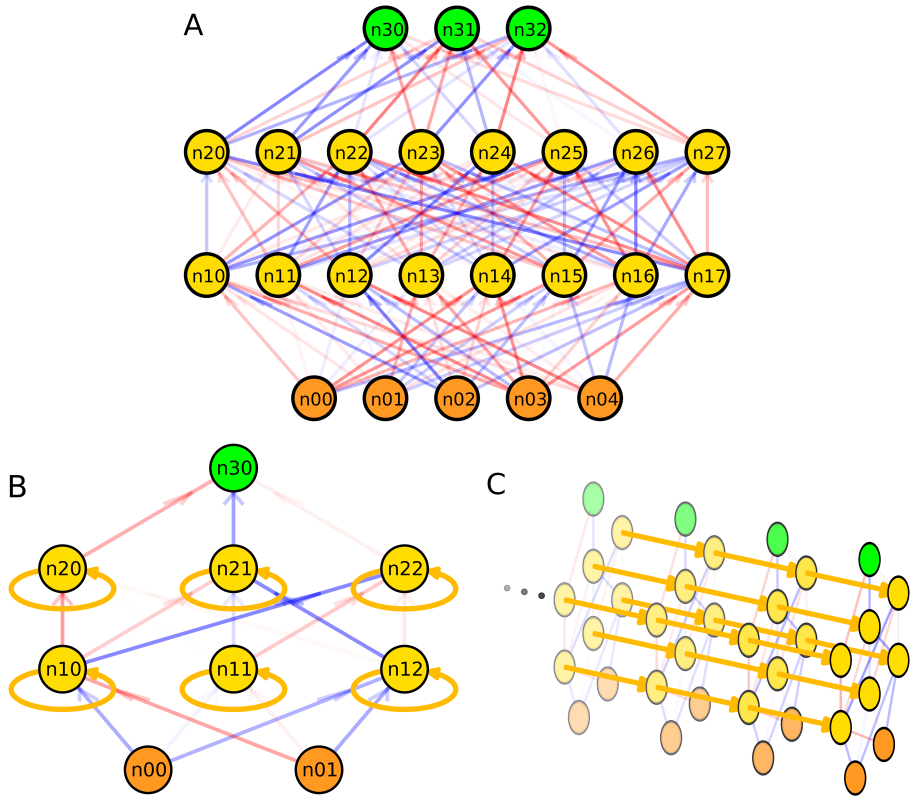


FIGURE 1.6: A: An example of a feed-forward neural network with 5 inputs (orange), two hidden layers of 8 neurons each (yellow) and 3 outputs (green). B: An example of a recurrent neural network, with two inputs (orange), two hidden layers of 3 neurons each (yellow) and one output (green). The recurrent connections are depicted with yellow. C: A possible interpretations of such a network that unrolls into several copies of itself. Such unrolling is typically considered due to computational practicality.

Typically, the neural dynamics are simplified such that inputs and outputs are scalar and all the inputs sum up and pass through an activation function. For example, the output of neuron $n13$ in Figure 1.6A is determined as:

$$n13 = \phi\Big(\sum_i n0i \times w0i13\Big)$$

where ϕ is its activation function and $w0i13$ the incoming weights from neurons $n0i$. The activation function introduces non-linearity, adding complexity to the system. Typical activation functions are sigmoids, "S" shaped functions starting at some value at $-\infty$ and continuously limiting at another value at ∞.

1.6.2 Recurrent neural networks

Recurrent neural networks are slightly more complicated because they incorporate connections that link states at a previous time with the current state. This makes them much more complex because they inherently posses memory. They have a forward oriented semantics which makes them a natural choice for time series prediction and they have had a lot of success in, e.g. audio and text prediction, translation and generation [48, 49]. A recurrent neural network is depicted in Figure 1.6B,C.

1.7 Thesis structure

This thesis consists of six chapters. The current first one as a general introduction, Chapters 2-4 correspond to three stand-alone scientific publications, Chapter 5 is a preliminary preprint, and lastly Chapter 6 as a general discussion.

In Chapter 2, I present a method of inferring dynamical properties from observations of a perturbed oscillator [50]. The oscillator is assumed to be self-sustained and weakly perturbed such that it follows the phase equation (6.4). Then, this phase equation is fitted to the data using an iterative procedure. For the data, a continuous observation of the perturbation signal is considered as well as time instants corresponding to the same value of the asymptotic phase of the oscillator. The method yields the natural frequency, the infinitesimal phase response curve as well as the asymptotic phase of the oscillator.

In Chapter 3, a similar phase inference as presented in Chapter 2 is extended to infer the connectivity of a network of oscillators [51]. The oscillators are again assumed to be self-sustained and weakly perturbed, following phase dynamics (6.4). Further, the coupling is assumed to be time localized to instances of spiking. From the spike trains of all units in the network I can infer the directed and weighted connectivity as well as the properties of individual oscillators: their natural periods and phase response curves.

In Chapter 4, I put the now trending neural networks to the test on inferring dynamics of different oscillatory systems [52]. I consider a scheme for training the networks on time series of a perturbed oscillatory system, and then estimate the desired dynamical properties from the trained network via an active experiment. Different properties can be inferred, including the phase response curve and the maximal Lyapunov exponent. I do this for several example oscillatory systems, including self-sustained, excitable and chaotic oscillators.

In Chapter 5, I apply the procedures of Chapter 4 to recordings of perturbed human gait. The spatial positions of several markers on the body are measured in time, from which I construct the state space of the dynamical system governing the gait pattern. Then, I train a neural network to reproduce its behaviour and extract the phase response curve by probing the inferred model.

Finally, I discuss the domain of applicability of the here-used methods and how they might behave in non-optimal settings, including unknown inputs, dynamics that do not fulfill chosen assumptions and insufficient amounts of data. Furthermore, I discuss the maximal theoretical content of data, elaborate on some particular choices made in designing the methods and outline their possible extensions.

Chapter 2

Inferring the phase response curve

Phase response curves are important for analysis and modeling of oscillatory dynamics in various applications, particularly in neuroscience. Standard experimental technique for determining them requires isolation of the system and application of a specifically designed input. However, isolation is not always feasible and we are compelled to observe the system in its natural environment under free-running conditions. To that end we propose an approach relying only on passive observations of the system and its input. We illustrate it with simulation results of an oscillator driven by a stochastic force.

Adapted from: R. Cestnik and M. Rosenblum, Inferring the phase response curve from observation of a continuously perturbed oscillator, *Scientific Reports* 8, 13606 (2018) [50].
Associated software: Phase reponse curve inference @ Software section at the end.

2.1 Introduction

Phase response curve (PRC), also known as phase resetting or phase sensitivity curve, is a basic characteristic of a limit cycle oscillator [16, 53, 54, 55]. This curve describes variation of the phase φ of the system in response to a weak external perturbation $p(t)$:

$$\dot{\varphi} = \omega + Z(\varphi)p(t) \ . \tag{2.1}$$

Here ω is the natural frequency and $Z(\varphi)$ is the oscillator's PRC. Thus, knowledge of $Z(\varphi)$ and ω completely determines the phase dynamics for any given weak perturbation. For example, it allows one to determine – at least numerically – the domain of locking to an external force of a given amplitude. Hence, techniques for efficient experimental determination of PRC are in high demand.

If it is possible to isolate the oscillator and apply a specially designed perturbation, then determination of PRC is straightforward. To this goal, the experimentalist has to "kick" the system with short and weak pulses at different values of the oscillator's phase and look for subsequent variation of one or several oscillation periods, i.e., for the evoked asymptotic phase shift [55, 53, 56, 57, 58]. If the oscillator is noisy – and all real-world systems are – then stimulation for each φ shall be performed many times and the results shall be averaged over the trials. Next, the whole procedure shall be repeated for different amplitudes of the stimulation in order to check whether the obtained PRC linearly scales with the amplitude: this would indicate that the chosen stimulation is sufficiently weak (we remind that in theory the PRC is defined for infinitesimally weak perturbations [54, 16]).

If one has no control over the applied input and instead has to rely on passive observations, the problem of invoking the PRC is in general not solved and the method will depend on the input. Possibly the best-case scenario is when the input is weak, spiky and with a frequency several times lower than that of the oscillator. Then, as long as the oscillator does not synchronize with the input, the spikes arrive at different phases and never more than one per period. This results in a focused perturbation which provides an estimate of phase shifts for different oscillator phases, like in the standard technique [55, 53, 56, 57, 58]. If, however, the frequency of the input spike train is higher than that of the oscillator then each period on average gets perturbed more than once, which considerably adds to the complexity of the problem [51]. In general the input may be arbitrary, in particular, it can be a continuous noise-like signal. If both such input and output of the system are measured, then one can infer the PRC using the idea of Spike-Triggered Average (STA) [59]. As has been demonstrated theoretically and numerically for the Hodgkin-Huxley neuronal model, for weak delta-correlated input the STA is proportional to the derivative of the PRC [59]. Recent more practical algorithms exploit weighted STA (WSTA): they imply rescaling of the input within each inter-spike interval to the same length and subsequent averaging with the weights, determined by the length of these intervals [60, 58]. Another approach is to use STA with a specific optimal colored noise input [61].

In this chapter we introduce a method for inferring the PRC by fitting the phase model (2.1) to the observed time series. This nonlinear problem is solved by an iterative procedure that provides frequency ω and PRC $Z(\varphi)$ as well as instantaneous phase $\varphi(t)$. An important novel feature of our algorithm is a built-in error estimation that allows us to monitor the goodness of the reconstruction. We demonstrate the

efficiency of the proposed method on several oscillator models driven by correlated noise and compare it with the techniques from Refs. [60, 58]. We show that our technique outperforms them in case of very short time series, not very short correlation and not very weak amplitude of the input.

2.2 Results

2.2.1 Technique

Suppose we perturb the oscillator under study with stimulation $p(t)$ and record its output $x(t)$. For example, $x(t)$ corresponds to the membrane potential of a cell. Since we assume that the oscillator has a stable limit cycle, without perturbation the process $x(t)$ would be periodic. In practice, because noise is inevitable, the oscillation would be almost periodic. The technique of extracting the PRC crucially depends on the properties of the signal $x(t)$ as well as those of the perturbation $p(t)$. We first discuss the relatively simple case where $x(t)$ is a smooth signal, suitable for phase estimation with the conventional technique of embedding the signal via the Hilbert transform [24, 62, 15] and performing the protophase-to-phase transformation [13].

Once an estimate of the phase $\varphi(t)$ is obtained, the instantaneous frequency $\dot{\varphi}(t)$ can be numerically estimated, and then, representing the PRC as a finite Fourier series,

$$Z(\varphi) = a_0 + \sum_{n=1}^{N} \left[a_n \cos(n\varphi) + b_n \sin(n\varphi) \right] \ , \tag{2.2}$$

we obtain from Eq. (2.1) a system of linear equations with respect to the unknown frequency ω and the Fourier coefficients a_n, b_n. Provided we have a good resolution and long enough time series, this system is overdetermined and can be solved by means of optimization, e.g., least squares minimization. Notice that this can be done for an arbitrary form of perturbation $p(t)$. Thus, a common approach to the problem revolves around estimating the phase φ and its derivative $\dot{\varphi}$ (the latter requires numerical differentiation and is therefore sensitive to noise). For cases when the signal has a complex form and there is no unique center of rotation in the Hilbert representation, alternative ways of estimating φ and $\dot{\varphi}$ shall be used. For example, for neural oscillations the phase is often obtained via simple linear interpolation between spikes [63, 60, 58]. In a different setup, like in studies of the interaction between the respiratory and cardiac oscillators several markers within each cycle of the electrocardiogram are used to determine the phase [14]. Such *ad hoc* techniques proved being useful, albeit they tend to be setup-specific, lack generality, and provide only zero-order approximation of the phase.

Here we introduce a technique that does not require computation of continuous phase φ and instantaneous frequency $\dot{\varphi}$ from the data. For our approach, in addition to continuous observation of the input forcing $p(t)$, it is sufficient to define one well-defined event within each cycle of $x(t)$ such that the times t_m of these events can be considered as the instants of return to a Poincaré surface of section. In the simplest case this event can correspond to a condition $x(t) = \text{const}$, while generally the problem of choosing a proper Poincaré section is not so trivial and is discussed in more detail below. Correspondingly, these events can be assigned the same value of the phase which without loss of generality can be set to zero, $\varphi(t_m) = 0$. For these

events one can choose spikes, threshold crossings, or even bursts (an example can be found in Supplementary material S2.1, Fig. S2.4). So, in analysis of spiking neurons it is natural to use spikes since they can be reliably detected while the signal between them can be dominated by noise. The phase increase within each inter-event interval, $T_m = t_{m+1} - t_m$, is defined to be 2π.

In what is the *first key step* of our approach, we integrate Eq. (2.1) over each interval T_m, replacing $Z(\varphi)$ by its Fourier representation via Eq. (2.2),

$$2\pi = \omega T_m + a_0 \int_{t_m}^{t_m+T_m} p(t)\mathrm{d}t + \sum_{n=1}^{N} \left[a_n \int_{t_m}^{t_m+T_m} p(t)\cos[n\varphi(t)]\mathrm{d}t + b_n \int_{t_m}^{t_m+T_m} p(t)\sin[n\varphi(t)]\mathrm{d}t \right] . \tag{2.3}$$

Certainly, the computation of the right hand side requires knowledge of the phase. Here we point out that in the limit of vanishing perturbations, the phase between events grows nearly linearly. Hence, for a sufficiently weak $p(t)$ the problem can be quite precisely solved by linearly interpolating the phase $\varphi(t) = 2\pi(t - t_m)/T_m$, inserting it in Eq. (2.3), numerically computing all the integrals, and solving the linear system by means of optimization, cf. [63]. However, for stronger driving amplitudes the increased inaccuracy of the linear approximation unavoidably translates to inaccuracy of the solution. We address this with the *second key step* in our technique, where the phase is estimated through an iterative procedure, by solving systems (2.1) and (2.3) in turns. For the zeroth approximation of the phase we linearly interpolate it between events, $\varphi^{(0)}(t) = 2\pi(t - t_m)/T_m$. Then, inserting it in Eq. (2.3) and solving the system we obtain the first approximation of the solution $\omega^{(1)}, Z^{(1)}$. The next approximation of the phase is obtained by numerically integrating Eq. (2.1), i.e., by solving $\dot{\varphi}^{(1)} = \omega^{(1)} + Z^{(1)}[\varphi^{(0)}(t)]p(t)$ separately for each inter-event interval, taking $\varphi^{(1)}(t_m) = 0$ for initial conditions. Since the phase $\varphi^{(0)}$, natural frequency $\omega^{(1)}$, and PRC $Z^{(1)}$ are known only approximately, the computed phase at the end of the interval, $\varphi^{(1)}(t_m + T_m) = \psi_m^{(1)}$, generally differs from the correct value 2π. Therefore, we *rescale* the phase within each interval: $\varphi^{(1)}(t) \to 2\pi\varphi^{(1)}(t)/\psi_m^{(1)}$. This rescaled phase is then used to obtain the second approximation of the solution $\omega^{(2)}$, $Z^{(2)}$, and then the whole procedure can be repeated to obtain further ones. As illustrated below, the iterations typically converge to the true solution and the quantities $\psi_m^{(k)}$ can be used to monitor the convergence.

2.2.2 Numerical tests

First we mention that for clarity of presentation we always set the natural frequency of the oscillator ω to 2π, thus ensuring the period of an unperturbed oscillator is $T = 1$ (with oscillators where natural frequency is not an explicit parameter this is done by rescaling time). Also, throughout this chapter, unless specified differently, we use $k = 10$ iteration steps, $N = 10$ Fourier harmonics, time of simulation $t_{\mathrm{sim}} = 500$ and time step $\mathrm{d}t = 0.001$.

Phase model: proof of concept

We first test the technique on models, where both the true phase and the PRC are known. For this purpose we simulate Eq. (2.1) with PRCs given as

$$Z(\varphi) = (1 - \cos(\varphi)) \exp\left(3[\cos(\varphi - \pi/3) - 1]\right) , \tag{2.4}$$

or

$$Z(\varphi) = -\sin(\varphi) \exp\left(3[\cos(\varphi - 0.9\pi) - 1]\right) . \tag{2.5}$$

These curves model PRC of type I and II respectively, as they are classified in the context of neuronal modeling [64, 51]. Type I curves are non-negative which means that every stimulus from an excitatory connection shortens the period duration, while type II curves are positive at some phases and negative at others meaning that a stimulus may shorten or lengthen the period depending on when it arrives. We take the input to the system to be an OrnsteinUhlenbeck process [26]

$$\dot{p} = -\frac{p}{\tau} + \varepsilon\sqrt{2/\tau}\ \xi(t) , \tag{2.6}$$

where ξ is Gaussian white noise $\langle \xi(t)\xi(t')\rangle = \delta(t-t')$, while ε and τ are the amplitude and correlation time of the driving signal, $\langle p(t)p(t')\rangle = \varepsilon^2\ e^{-\frac{|t-t'|}{\tau}}$. At this point we mention that for the same value of ε different oscillators are perturbed by different amounts because they have different PRCs and frequencies. From Eq. (2.1) we see that the effect of perturbation is proportional to $\varepsilon\|Z\|/\omega$ where $\|\cdot\|$ stands for norm (we use the L_2 functional norm, see Methods section 2.4). Therefore, to compare the results for different test models and since all frequencies are set to 2π, we specify driving strength with the quantity $\varepsilon\|Z\|$.

Spike data is generated using the condition $\varphi(t_m) = m \cdot 2\pi$. Then, we perform k iterations of the reconstruction procedure and compare the results against the data. To quantify the quality of the reconstruction we compute the difference between the recovered and true PRCs, Δ_Z, and average deviation of ψ_m from 2π, Δ_ψ (see Eqs. (2.10) and (2.12) in Methods section 2.4). Δ_Z should be compared to 1, where $\Delta_Z \ll 1$ means the reconstructed PRC matches the true one closely, while $\Delta_Z \approx 1$ signals that the reconstructed PRC does not resemble the true one at all. Δ_ψ is to be compared to a related measure of event irregularity, Δ_{ψ_T} (see Eq. (2.13) in Methods section 2.4), where $\Delta_\psi \ll \Delta_{\psi_T}$ means that our reconstruction can reproduce the data well, while $\Delta_\psi \approx \Delta_{\psi_T}$ signals that, statistically speaking, the reconstructed PRC and frequency are not able to reproduce the data any more accurately than a perfectly periodic oscillator with frequency $\langle\omega\rangle = \langle\frac{2\pi}{T_m}\rangle$.

In Fig. 2.1 we show a reconstruction example for the first test. We use fairly strong driving $\varepsilon\|Z\| = 5$ and moderate correlation $\tau = 0.1$ so that the phase alterations are visible in the plot. Notice that the error measures indicate a good reconstruction, i.e., $\Delta Z \ll 1$ and $\Delta_\psi \ll \Delta_{\psi_T}$. Notice also that the initial PRC estimate $Z^{(1)}$ is quite imprecise while further iterations improve in precision and converge to the true curve. To see this convergence in more detail we plot the error measures for each iteration k in Fig. 2.2 (a, b), for PRC type II. We also investigated how the error depends on the duration of data recordings t_{sim} and found that a few hundred periods suffice for a reliable reconstruction, see Fig. 2.2 (c, d).

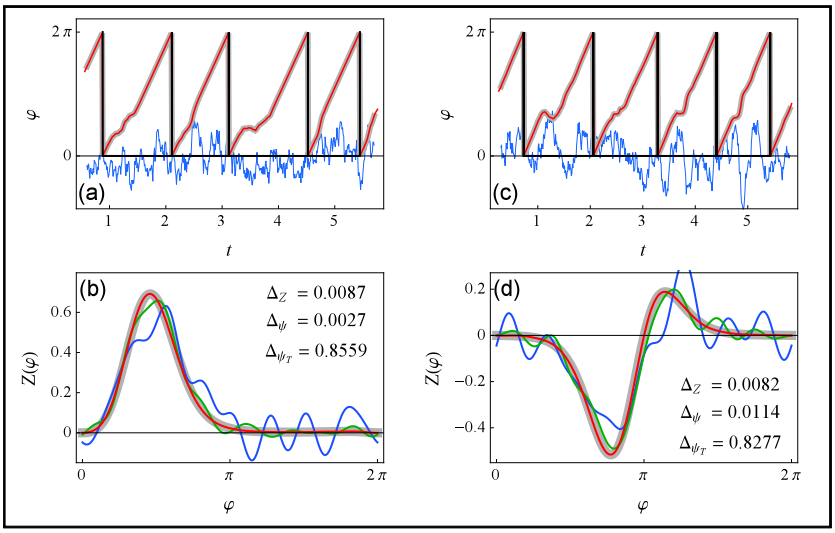


FIGURE 2.1: A reconstruction of phase φ and PRC Z for PRC type I (a, b) and type II (c, d). In the top plots (a, c), true phase is plotted in thick gray, reconstructed phase in red, driving signal in blue (scaled for being visually comparable to the phase) and times of spikes t_m in black. In the bottom plots (b, d), true PRC is plotted in thick gray, first iteration $Z^{(1)}$ in blue, second $Z^{(2)}$ in green and tenth $Z^{(10)}$ in red. Parameters are $\varepsilon\|Z\| = 5$ and $\tau = 0.1$.

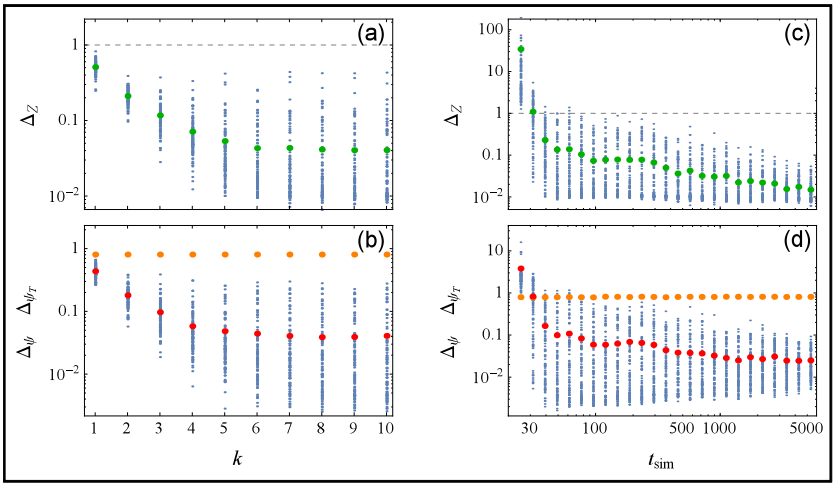


FIGURE 2.2: Errors of reconstruction Δ_Z and Δ_ψ, Eqs. (2.10) and (2.12), versus iteration number k (a, b) and versus the duration of data recordings $t_{\rm sim}$ (c, d), for PRC type II. For each k and each $t_{\rm sim}$, 100 points corresponding to different realizations of noise are plotted with blue dots and their average in colors: Δ_Z in green, Δ_ψ in red (and Δ_{ψ_T} in orange for reference). Parameters are $\varepsilon\|Z\| = 5$, $\tau = 0.1$, $t_{\rm sim} = 500$ (a, b) and $k = 10$ (c, d).

The Morris-Lecar neuron: effect of amplitude ε and correlation time τ

Our next test model is the Morris-Lecar neuronal oscillator [65, 54]

$$\begin{aligned} \dot{V} &= I - g_L\,(V - V_L) - g_K\,w\,(V - V_K) - g_{Ca}\,m_\infty(V)\,(V - V_{Ca}) + p(t)\,, \\ \dot{w} &= \lambda(V)\,(w_\infty(V) - w)\,, \end{aligned} \tag{2.7}$$

where

$$\begin{aligned} m_\infty(V) &= [1 + \tanh((V - V_1)/V_2)]/2 \,, \\ w_\infty(V) &= [1 + \tanh((V - V_3)/V_4)]/2 \,, \\ \lambda(V) &= \cosh[(V - V_3)/(2V_4)]/3 \,, \end{aligned} \tag{2.8}$$

and the parameters are: $I = 0.07$, $g_L = 0.5$, $g_K = 2$, $g_{Ca} = 1.33$, $V_1 = -0.01$, $V_2 = 0.15$, $V_3 = 0.1$, $V_4 = 0.145$, $V_L = -0.5$, $V_K = -0.7$ and $V_{Ca} = 1$. A reconstruction depiction similar to Fig. 2.1 can be found in Supplementary material S2.1, Fig. S2.9.

We investigate how amplitude of driving ε effects the reconstruction. For that purpose we simulate a forced oscillator with different values of ε and compute the errors of reconstruction Δ_Z, Eq. (2.10), and Δ_ψ, Eq. (2.12), see Fig. 2.3. The true PRC of the Morris-Lecar system is obtained via the standard technique, see Methods section 2.4 and Ref. [58]. For weak driving the PRC error Δ_Z is small and independent of amplitude, while Δ_ψ and Δ_{ψ_T} scale approximately linearly with amplitude and maintain a constant ratio, $\Delta_\psi/\Delta_{\psi_T} \approx \text{const} \ll 1$. The reconstruction works well, as expected. As the driving amplitude increases, at some point Δ_Z begins to grow and Δ_ψ starts to approach Δ_{ψ_T}, a sign of declining reconstruction quality (in Fig. 2.3 this happens around $\varepsilon\|Z\| \approx 5$). In this regime the distribution of errors (both Δ_Z and Δ_ψ) widens, meaning that reconstruction quality for the same parameters may vary considerably from trial to trial. As the driving gets very strong (around $\varepsilon\|Z\| \approx 100$ in Fig. 2.3) the reconstruction does not work, i.e., $\Delta_Z \approx 1$ and $\Delta_\psi \approx \Delta_{\psi_T}$. As a side note, in Fig. 2.3 (b) one can observe what looks like a change in behavior around $\varepsilon\|Z\| \approx 100$, where there is a jump in Δ_ψ and Δ_{ψ_T}. This is due to the test model we use, Morris-Lecar, and its behavior when strongly stimulated (at that amplitude the system spends some time in the vicinity of the fixed point) and is not a consequence of our technique (in Supplementary material S2.1 one can see analogous plots for other oscillators where this is not observed, Fig. S2.9).

In the same way as for the amplitude, we now investigate how correlation time τ effects the reconstruction, see Fig. 2.4. The reconstruction works best for short correlation times: Δ_Z as well as the ratio $\Delta_\psi/\Delta_{\psi_T}$ are small and roughly constant, but in this case Δ_ψ and Δ_{ψ_T} grow sublinearly with ε. Similarly to what we have seen in the previous figure, as we increase τ there comes a point where Δ_Z starts to grow and Δ_ψ starts to approach Δ_{ψ_T} (this happens around $\tau \approx 0.4$ in Fig. 2.4). Likewise the distribution from this point on widens, indicating that reconstruction quality varies from trial to trial. As we continue to increase τ there comes a point at which average Δ_Z reaches 1 and Δ_ψ reaches Δ_{ψ_T} (this happens around $\tau \approx 30$ in Fig. 2.4). Interestingly, we can see that while the error averages in general reflect a bad reconstruction, there persists a branch of reconstructions that maintains a low error which is even slightly decreasing with τ. This means that with a small probability we can still get a good reconstruction from slow varying input (an example can be found in Supplementary material S2.1, Fig. S2.3), and by monitoring the solution errors we can distinguish successful reconstructions from the unsuccessful ones. A similar qualitative description is valid for other oscillators in this context as well (analogous analysis on other oscillators can be found in Supplementary material S2.1, Fig. S2.9). Here it is worth mentioning that while our chosen input, Eq. (2.6), yields white noise with zero intensity in the limit $\tau \to 0$, we have tested our procedure for white noise with finite intensity and it works as well (an example can be found in Supplementary material S2.1, Fig. S2.1).

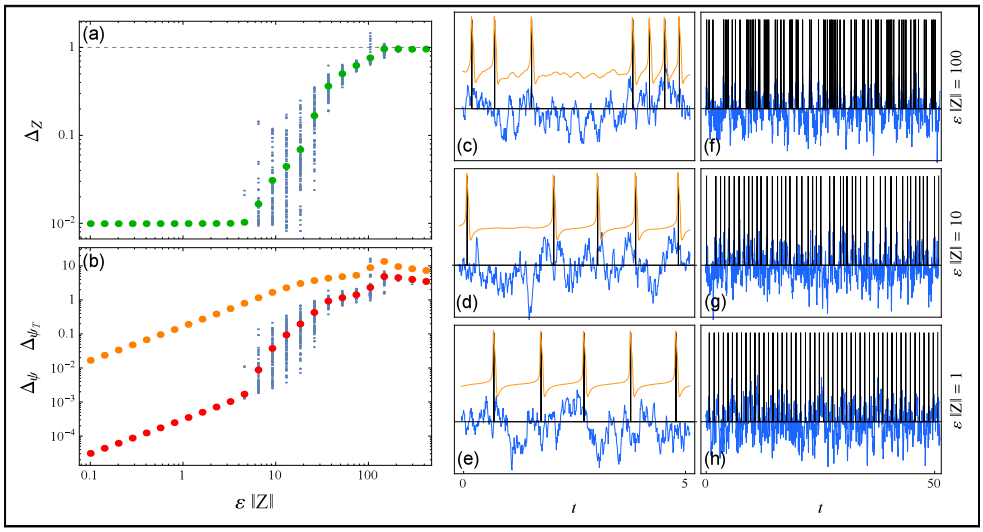


FIGURE 2.3: The dependence of errors of reconstruction Δ_Z and Δ_ψ, Eqs. (2.10) and (2.12), on driving amplitude ε for the Morris-Lecar neuronal oscillator Eq. (3.10). In (a, b) for each value of ε, 100 points corresponding to different realizations of noise are plotted with blue dots and their average in colors: Δ_Z in green, Δ_ψ in red (and Δ_{ψ_T} in orange for reference). Shown in (c – h) are segments of signal on two time scales for three amplitudes of driving $\varepsilon\|Z\|$. In orange the voltage $V(t)$ (only in (c – e)) and in blue the input signal $p(t)$, scaled for being visually comparable. The times of zero phase t_m are marked with vertical black lines. The correlation time is $\tau = 0.1$.

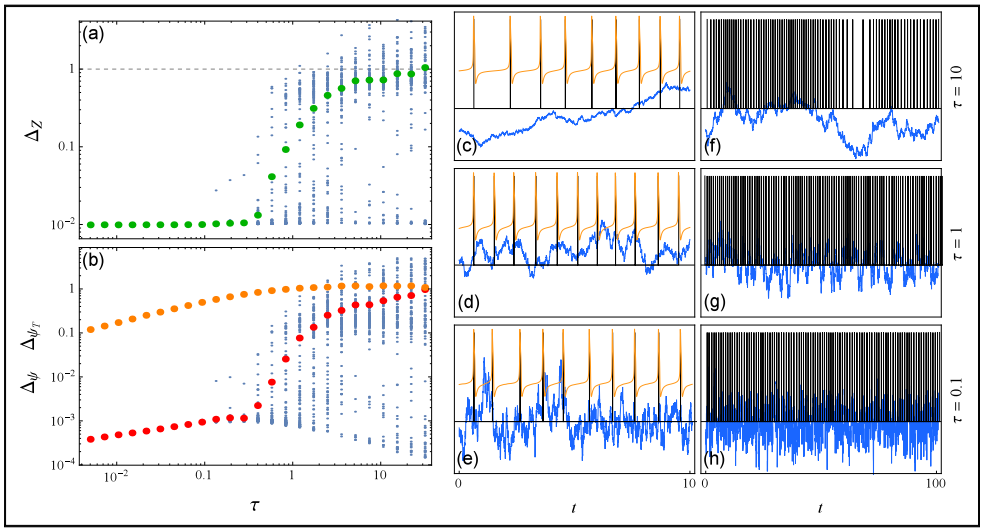


FIGURE 2.4: The dependence of errors of reconstruction Δ_Z and Δ_ψ, Eqs. (2.10) and (2.12), on correlation time τ of the driving for the Morris-Lecar oscillator, Eq. (3.10). In (a, b) for each value of τ, 100 points corresponding to different realizations of noise are plotted with blue dots and their average in colors: Δ_Z in green, Δ_ψ in red (and Δ_{ψ_T} in orange for reference). Shown in (c – h) are segments of signal on two time scales for three values of correlation time τ. In orange the voltage $V(t)$ (only in (c – e)) and in blue the input signal $p(t)$, scaled for being visually comparable. The times of zero phase t_m are marked with vertical black lines. The driving strength is $\varepsilon\|Z\| = 3$.

Up to this moment we have been postponing a discussion of a crucial problem, namely how to determine the instants t_m corresponding to zero phase, $\varphi(t_m) = 0$.

For spiky data where only the times of events can be reliably detected (so that the data is represented by a point process) there is no other option but to use the times of events. This corresponds to pure phase models, where this problem does not exist and the values t_m are obtained in the course of simulation. This is not the case for full models, where both phases and amplitudes shall be taken into account. So, for the analysis of the Morris-Lecar model we used a simple threshold rule and defined zero phase from the condition $x(t_m) = x_{\min} + 0.9(x_{\max} - x_{\min})$, $\dot{x}(t_m) < 0$, where $x_{\min}$ and $x_{\max}$ are the minimal and maximal measured value of the observed signal. However, this is not exact: in fact, the proper section of the limit cycle shall coincide with a line (or, generally, surface) of constant phase, called isochrone [66], but in experimental conditions the model equations and isochrones are not known *a priori*. Nevertheless, for neuronal oscillators this is not a complication if the threshold is chosen close to the maximum of the pulse, i.e., in the domain of fast motion along the limit cycle. Here the density of isochrones is low, i.e., phase gradient in the direction perpendicular to the isochrones is small (for an illustration of the isochrone structure see Supplementary material S2.1, Fig. S2.6b). Respectively, the error due to deviation of the line $x = \text{const}$ from the isochrone is small as well. For general oscillators this is not true, and the problem requires a special treatment, illustrated by the following example.

The van der Pol oscillator: importance of choosing a proper Poincaré section

Our third test system is the van der Pol oscillator [67]

$$\ddot{x} - 2(1 - x^2)\dot{x} + x = p(t) \; . \tag{2.9}$$

With this model we address the importance of choosing an appropriate Poincaré section for determination of the beginnings of each cycle, i.e., instants t_m corresponding to zero phase $\varphi(t_m) = 0$. Here we make use of the ability of our technique to monitor the error of reconstruction.

As a first step we probe different threshold values for the signal $x(t)$ and choose the one which yields the smallest error Δ_ψ. For this purpose we parametrize the threshold values with a parameter $0 < \theta < 1$, see Methods section 2.4 for details, and perform a search over θ. For each value of the threshold θ, 100 simulations with different noise realizations were performed. We have calculated both the error of the PRC and the Δ_ψ error. We underline, that Δ_ψ can be calculated from data alone, without any *a priori* knowledge of the system. The results are shown in Fig. 2.5. A clear minimum in both errors around the value $\theta = 0.7$ indicates the optimal threshold that corresponds to the surface of section tangential to the local isochrone, see Fig. 2.6b.

Generally, the goodness of the surface of section (a line in this example) is determined by two factors: the line should be tangential to an isochrone and it should be chosen in the domain of fast motion. This cannot always be ensured by simple thresholding, $x(t_m) = \text{const}$. Therefore, we consider inclined lines of section, corresponding to local linear approximation of isochrones. This can be done even if we have access to only one variable, in our case $x(t)$. As is well-known, a phase portrait, topologically equivalent to the true one, can be obtained from a single time series via computation of the derivative or time-delayed embedding [68, 69]. We use here

the former option, computing $\dot{x}(t)$ using the five point finite difference scheme [70]. In the two-dimensional embedding with coordinates $(x, \dot{x})$ we then search for an optimal linear section, determined by its position θ and inclination α, see Methods section 2.4 for details. We use a single simulated time series with the total length of 500 periods, mimicking a real world scenario with limited collected data available, and compute the error Δ_ψ, for each pair of values θ, α. In this way a possibly better surface of section is found, see Fig. 2.6. This approach can be further improved by a high-dimensional search, e.g., by locally approximating isochrones with a higher order polinomial.

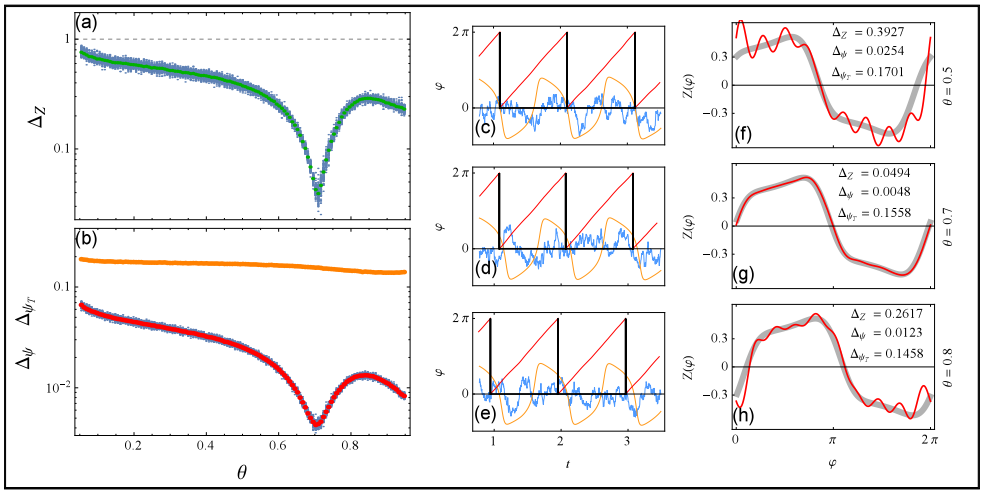


FIGURE 2.5: The dependence of errors of reconstruction Δ_Z and Δ_ψ, Eqs. (2.10) and (2.12), on the chosen Poincaré section, determined by the threshold parameter θ (see Methods section 2.4), for the van der Pol oscillator, Eq. (2.9). In (a, b) for each value of θ, 100 points corresponding to different realizations of noise are plotted with blue dots and their average in colors: Δ_Z in green, Δ_ψ in red (and Δ_{ψ_T} in orange for reference). Shown in (c – e) are segments of signal for three values of θ. Variable $x(t)$ in orange and the input $p(t)$ in blue, scaled for being visually comparable. The times of zero phase t_m are marked with vertical black lines and in red is the reconstructed phase. In (f – h) the corresponding PRC reconstructions are plotted with red. The true PRC is depicted with a thick gray curve. Parameters are $\varepsilon\|Z\| = 1$ and $\tau = 0.1$.

Comparison with the WSTA technique

In the last test we compare the performance of our approach with that of the WSTA method [60, 58], see Fig. 2.7. As expected, our technique performs better if the correlation time and/or amplitude of the input is relatively large. This is due to the fact that, in contradistinction to WSTA, we do not use the assumptions of delta-correlated input and of linearly growing phase. Moreover, the test demonstrates that our technique works better for shorter time series.

2.3 Discussion

To summarize, in this chapter we introduce a new method for obtaining the PRC and natural frequency of an oscillator relying only on observation of the oscillator's signal and its continuous perturbation. We demonstrate its efficiency by recovering

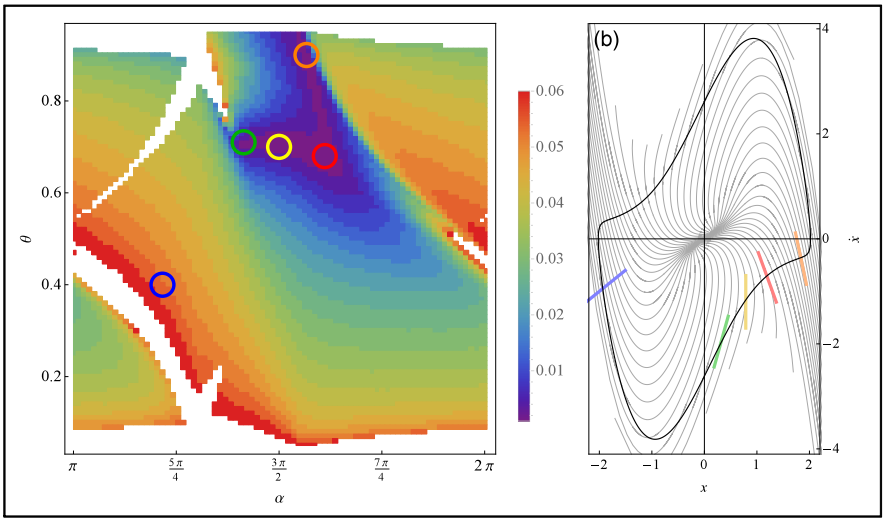


FIGURE 2.6: The error of reconstruction Δ_ψ, Eq. (2.12), with respect to the inclination of the Poincaré surface of section α, and its relative position θ, see Methods section 2.4 for the definition. In (a) the white regions correspond to error values out of range $\Delta_\psi > 0.06$. The particular values marked with colored circles correspond to example surfaces of section plotted in (b) with lines of the same color. The red ($\Delta_\psi = 0.0045$), green ($\Delta_\psi = 0.0051$), yellow ($\Delta_\psi = 0.0049$) and orange ($\Delta_\psi = 0.0051$) sections yield good period determination: the error is low and in (b) we see that they have a similar inclination to the local isochrones, plotted with thin gray curves. The blue one ($\Delta_\psi = 0.054$) is an example of an inaccurate surface of section that does not correspond to local isochrones (see (b)) and therefore yields a high error. Notice that the yellow line corresponds to the optimal section in Fig. 2.5. The search was performed using a single simulation run of length $t_{\mathrm{sim}} = 500$ (corresponding to roughly 500 periods). Parameters are $\varepsilon\|Z\| = 1$ and $\tau = 0.1$.

the PRC of several model systems driven by correlated noise, from only a few hundred observed periods. Furthermore, we provide error measures indicating the quality of the inference that can be calculated from data, without any additional knowledge of the system.

We then explore the effects of the amplitude and correlation time of the driving signal. Generally, the reconstruction works best for perturbation with low amplitude and short correlation time, although the requirements here are not so crucial as in the case of the WSTA technique [60, 58]. The degradation of the model inference at higher amplitudes can be due to two reasons: the phase description of an oscillator in terms of Eq. (2.1) loses its validity or/and linear approximation of the phase dynamics becomes too poor for the iterative scheme to converge. The latter problem may be solved by using a different initial estimate of the phase, e.g., obtained via the Hilbert transform. On the other hand, the degradation of the reconstruction procedure at large correlation times occurs due to a limitation of our technique. Indeed, if the driving is very slow and can be approximately considered as constant within one oscillation period, then the integrals in Eq. (2.3) vanish and the whole procedure fails. However, for some realizations of noise, relatively strong and slow driving yields a good reconstruction as well (see examples in Supplementary material S2.1, Fig. S2.2 and Fig. S2.3) and the introduced error measures allow us to examine whether the reconstruction is good or not. Finally, we mention that in the white noise limit an additional term, proportional to the square of noise intensity, appears in Eq. (2.1) [71]. This fact can be another source of error for large perturbation amplitudes.

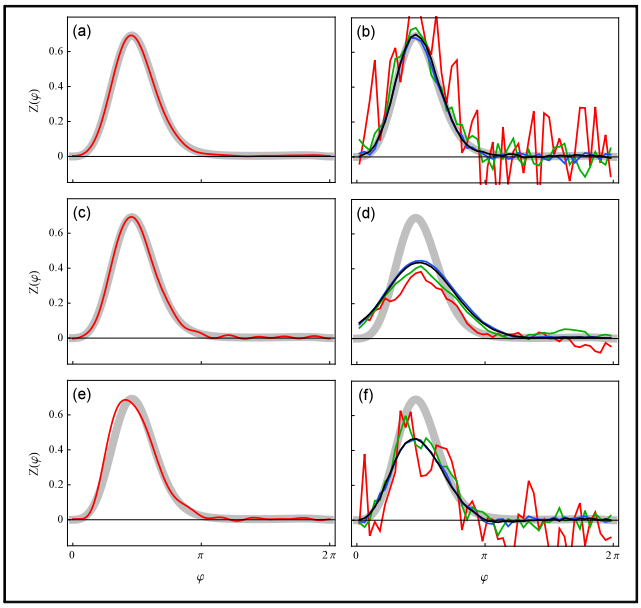


FIGURE 2.7: Comparison of our method (a, c, e) and the WSTA method [60, 58] (b, d, f). The data was simulated using the phase model (2.1). Different colors correspond to the length of time series used for the reconstruction: $t_{\mathrm{sim}} = 10^2$ in red, 10^3 in green, 10^4 in blue and 10^5 in black (for our method (a, c, e) only the $t_{\mathrm{sim}} = 10^2$ is plotted). The true PRC is depicted with a thick gray curve. In (a, b) the coupling is relatively weak, $\varepsilon\|Z\| = 5$, and the correlation time of the input is relatively small, $\tau = 0.01$. These are good conditions for both methods, and indeed both perform well. In (c, d) the coupling remains as before but the correlation time is larger, $\tau = 0.1$. In (e, f) the coupling strength is increased, $\varepsilon\|Z\| = 20$, while the correlation time is small again, $\tau = 0.01$.

Finally, we analyze the importance of proper determination of the points that can be considered as beginning of the periods, i.e., the points that are assigned zero phase. This tends to be straightforward for most neuronal models where spikes are the natural choice both due to their detection being robust to noise and the fact that during a spike the phase gradient is typically small dismissing the importance of the shape of the chosen surface of section. In general though, the choice of an appropriate surface of section is crucial. To this end we propose to search for an optimal Poincaré section which minimizes the reconstruction error Δ_ψ. For this search we parameterize these sections by two quantities, namely their position and inclination, use a two-dimensional embedding of the measured time series, and vary these quantities to find the combination that yields minimal Δ_ψ. In this way we find local approximations of the isochrones by straight lines (cf Fig. S2.8 in Supplementary material S2.1). This approach can be further extended to a nonlinear fit of isochrones. For high-dimensional systems the performance of the PRC estimation can be further improved by using a three-dimensional embedding and approximating the isochrones by inclined planes, etc. Alternatively, for a long data set one can reveal the isochrone structure by estimating the surfaces with the mean first return time equal to the period [9]. Next, our approach can be also combined with the technique suggested in the framework of WSTA in Ref. [58]. There the authors addressed the problem of choosing the proper section by rescaling and averaging $n > 1$ consecutive periods instead of one. The inaccuracy due to deviation of the used Poincaré section from the true isochrone is then distributed over n periods. For our approach the inter-event

intervals $T_m = t_{m+1} - t_m$ shall be simply replaced by $T_m^{(n)} = t_{m+n} - t_m$, and the left hand side of Eq. (2.3) shall be changed to $n \cdot 2\pi$. However, other numerical errors accumulate, and therefore we do not expect this approach to be superior than the search for an optimal section.

2.4 Methods

2.4.1 Reconstruction errors

For quantification of how well our reconstructed PRC resembles the true one we use the following expression

$$\Delta_Z = \left(\| Z^{(\mathrm{t})} - Z^{(\mathrm{r})} \| \right) / \| Z^{(\mathrm{t})} \| \,, \tag{2.10}$$

where $Z^{(\mathrm{t})}$ refers to the true PRC, $Z^{(\mathrm{r})}$ to the reconstructed one and $\|\cdot\|$ is the L_2 function norm: $\|f\| = \left[\int_0^{2\pi} f^2(\varphi) \mathrm{d}\varphi \right]^{1/2}$. When the reconstructed PRC nearly coincides with the true one the error is small, i.e., $\Delta_Z \ll 1$. When the two PRCs are not similar, but are of the same order of magnitude, Δ_Z is of the order of 1, but in general Δ_Z can have arbitrarily large values. For computation of the true PRC we first determine the period of the autonomous limit cycle oscillation, T_0, using the Henon trick [11]. Next, similarly to [58], we instantaneously perturb the oscillator at different phases within one period (phases are taken proportionally to the time from the beginning of the cycle) and look for the shift in the asymptotic phase. If the period where perturbation arrives is denoted by T_1 and the following periods as T_2, T_3, ..., then the phase shift normalized to the perturbation strength ε is

$$Z^{(\mathrm{t})}(\varphi) = 2\pi \frac{nT_0 - \sum_{i=1}^{n} T_i}{\varepsilon T_0} \,. \tag{2.11}$$

The number of periods n required for a complete relaxation to the limit cycle can be easily chosen by trial and error by checking that the result does not depend on n.

Next, we quantify how well our reconstruction describes the system which generated the data. ψ_m is defined as the reconstructed phase at the end of interval m which in general differs from the true phase of 2π due to the inaccuracy of the reconstructed PRC and ω. Therefore the average deviation of ψ_m from 2π is a natural measure of goodness for our reconstruction

$$\Delta_\psi = \left\langle \left(\psi_m - 2\pi \right)^2 \right\rangle^{1/2} \,, \tag{2.12}$$

where $\langle \cdot \rangle$ refers to averaging over m. It can be computed solely from data (unlike Δ_Z which requires the knowledge of the true PRC) meaning that it can be used in real experiments. However, Δ_ψ should not be taken as an absolute measure because it also depends on the inherent irregularity of inter-event intervals. With that in mind we define a measure of event irregularity

$$\Delta_{\psi_T} = \left\langle \left(\langle \omega \rangle T_m - 2\pi \right)^2 \right\rangle^{1/2} \,, \tag{2.13}$$

where $\langle\omega\rangle = \langle\frac{2\pi}{T_m}\rangle$ is the mean frequency. This measure serves as a good benchmark since it can be understood as Δ_ψ calculated for a trivial reconstruction, namely, when average period is used as a prediction of next inter-event interval. Hence, an indication of good reconstruction is the condition $\Delta_\psi \ll \Delta_{\psi_T}$.

2.4.2 A choice of Poincaré section

In the simplest case the same phase is assigned to the instants when the signal crosses a chosen threshold from above (the crossings from bellow can be taken as well so long as we are consistent). Practically, this instants are determined by linear interpolation between the closest values above and below the threshold. In this work we parameterize a threshold value s_{thr} by $\theta \in (0, 1)$, so that $s_{\text{thr}} = s_{\min} + \theta(s_{\max} - s_{\min})$, where $s_{\min}$ and $s_{\max}$ are the minimum and maximum of the considered signal. With the embedded signal, $x(t)$, $\hat{x}(t)$, we choose a Poincaré section as a straight line at an angle α (in our case we use $\hat{x} = \dot{x}$). This is accomplished by rotating the embedded limit cycle by the angle α, such that the line of section becomes parallel to the horizontal axis and then taking the vertical axis projection of the limit cycle as an auxiliary signal: $s_{\text{aux}} = -x(t)\sin(\alpha) + \hat{x}(t)\cos(\alpha)$. Then the auxiliary signal is thresholded as before. A depiction of the process may be found in Supplementary material S2.1, Fig. S2.7.

Author contributions statement

R.C. and M.R. designed the algorithms and performed the computations. R.C. computed the statistics and prepared final figures. Both authors wrote the manuscript.

Acknowledgements

We thank Arkady Pikovsky, Igor Goychuk and Chunming Zheng for useful discussions. The work was supported by ITN COSMOS (funded by the European Union Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 642563). Numerical part of this work conducted by M.R. was supported by the Russian Science Foundation (Project No. 14-12-00811).

S2.1 Supplementary material

S2.1.1 White noise driving

Here we demonstrate a reconstruction with Gaussian white noise input $\langle p(t)p(t')\rangle = \varepsilon^2\delta(t - t')$, see Fig. S2.1. We simulate the system by integrating Eq. (2.1) in the main text with the Milstein scheme [72] using the PRCs introduced in the main text, Eqs. (3.1) and (3.2). We record the differences of the Wiener process ΔW as the driving signal.

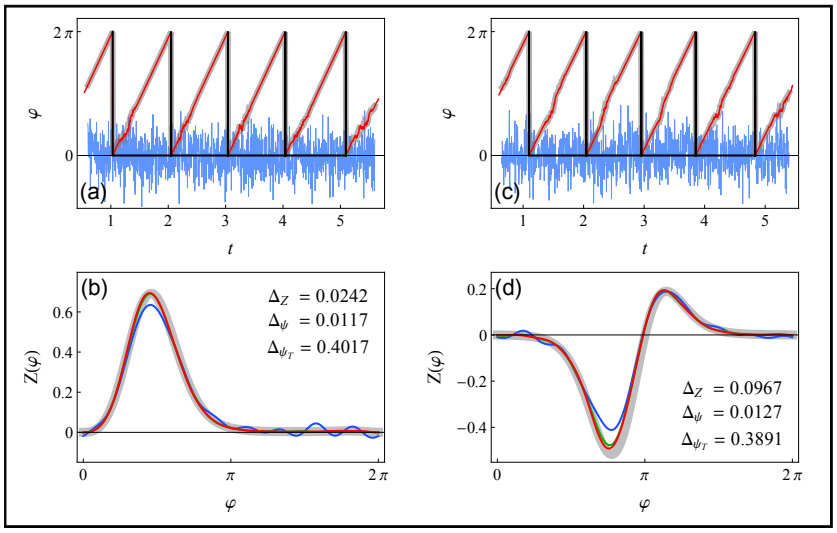


FIGURE S2.1: A reconstruction for a phase oscillator driven by white noise (analogous to Fig. 2.1 in the main text, (a, c) in blue differences of the Wiener process ΔW instead of signal p). The driving strength is $\varepsilon\|Z\| = 1$.

S2.1.2 The Morris-Lecar neuron: examples of strong: $\varepsilon\|Z\| = 20$, and slow: $\tau = 10$ driving

On average, cases with strong and slow driving yield bad reconstructions, however, this can vary from case to case. Here we show examples of good reconstructions for those circumstances, see Fig. S2.2 for strong and Fig. S2.3 for slow driving. They were chosen as the best from 20 trials (in terms of error measures, Eqs. (2.10), (2.12) and (2.13) introduced in the main text).

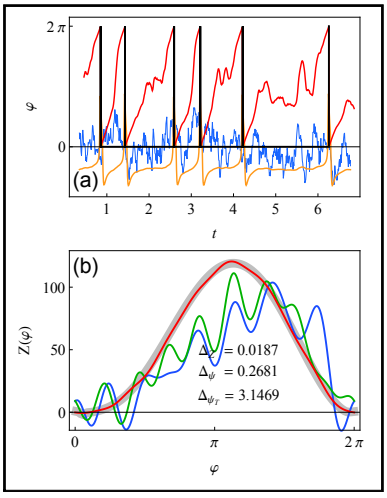


FIGURE S2.2: An example of a reconstruction with strong driving, Morris-Lecar neuron (similar to Fig. 2.1 in the main text). Parameters are $\varepsilon\|Z\| = 20$ and $\tau = 0.1$.

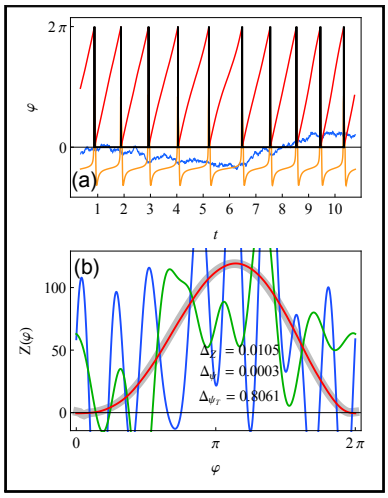


FIGURE S2.3: An example of a reconstruction with slow driving, Morris-Lecar neuron (similar to Fig. 2.1 in the main text). Parameters are $\varepsilon\|Z\| = 3$ and $\tau = 10$.

S2.1.3 The Hindmarsh-Rose neuron: example on bursting

To further support the argument that the data our method requires is general we perform a reconstruction on a bursting neuron. We use the Hindmarsh-Rose neuronal model [73]:

$$\begin{aligned} \dot{x} &= I + y - ax^3 + bx^2 - z + p(t) \,, \\ \dot{y} &= c - dx^2 - y \,, \\ \dot{z} &= r[s(x - x_R) - z] \,, \end{aligned} \tag{S14}$$

the parameters are: $I = 1.28$, $a = 1$, $b = 3$, $c = 1$, $d = 5$, $r = 0.0006$, $s = 4$ and $x_R = -1.6$. In this regime the neuron is bursting periodically. For the reconstruction we use parameters $\varepsilon\|Z\| = 0.01$, $\tau = 0.01$, $t_{\text{sim}} = 1000$, $\mathrm{d}t = 0.0001$ and $N = 15$. The beginning of the period is determined by the time of the first spike in a burst (this is arbitrary, any of the spikes could be considered for this role). See Fig. S2.4 for a reconstruction depiction. The PRC of this bursting cycle has a very sharp feature around the phase $\varphi \approx 1$ where the bursting stops. This is hard to capture with our method since we take the Fourier representation of the PRC and steep features require several harmonics to be expressed. Taking many harmonics can be impractical since each harmonic yields an unknown constant needed to be optimized, which can make the data requirements considerably big. With $N = 15$ Fourier harmonics that feature is to some degree smeared in phase, but still apparent.

S2.1.4 The van der Pol oscillator: varying the number of Fourier harmonics N

We explore the effect of using different numbers of Fourier harmonics N. The PRC of the oscillator for the chosen parameters cannot be captured well by a single harmonic term but we also know that using more terms means more unknowns to fit and therefore a necessity for more data to constrain them. By increasing the number of

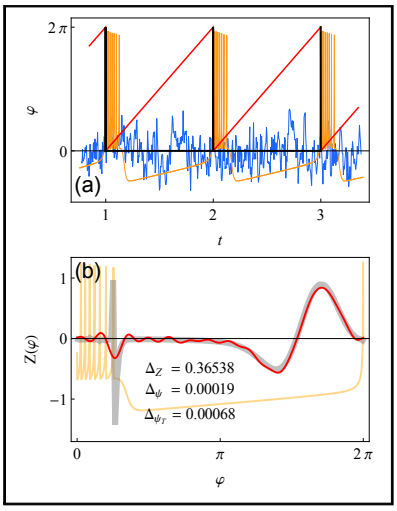


FIGURE S2.4: An example of a reconstruction using bursts as the events with equal phase, Hindmarsh-Rose neuron (similar to Fig. 2.1 in the main text). Parameters are $\varepsilon\|Z\| = 0.01$ and $\tau = 0.01$.

Fourier harmonics N while keeping the length of the time series t_{sim} constant we therefore expect the error to have a minimum, see Fig. S2.5.

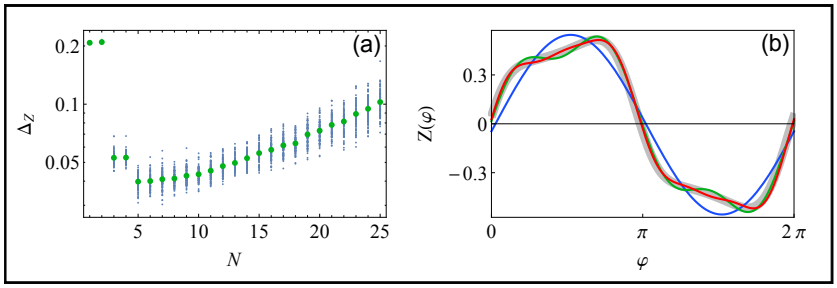


FIGURE S2.5: The dependence of PRC error Δ_Z, Eq. (2.10) in the main text, on the number of Fourier harmonics N for the van der Pol oscillator Eq. (2.9) in the main text. In (a) for each value of N, 100 points corresponding to different realizations of noise are plotted with blue dots and their average in green. In (b) PRCs reconstructed using $N = 1$, 3 and 5 in blue, green and red respectively (true PRC in thick gray). Parameters are $\varepsilon\|Z\| = 1$ and $\tau = 0.1$.

S2.1.5 Choice of a proper Poincaré section

In the case of a forced oscillator, the correct Poincaré section for determining instants of same phase corresponds to an isochrones surface. Here we show the isochrone structure of the two oscillators used in the main text, van der Pol and Morris-Lecar, see Fig. S2.6. The isochrones were computed using the backward integration method [7].

Now suppose we only measure one variable $x(t)$ but want to embed our signal in 2 dimensions and then determine the periods using a Poincaré section at an arbitrary angle α. This is done by rotating the embedded limit cycle and calculating the threshold value accordingly, a depiction can be seen in Fig. S2.7. Using a similar search to that performed in the main text in Fig. 2.6, we can estimate the isochrone

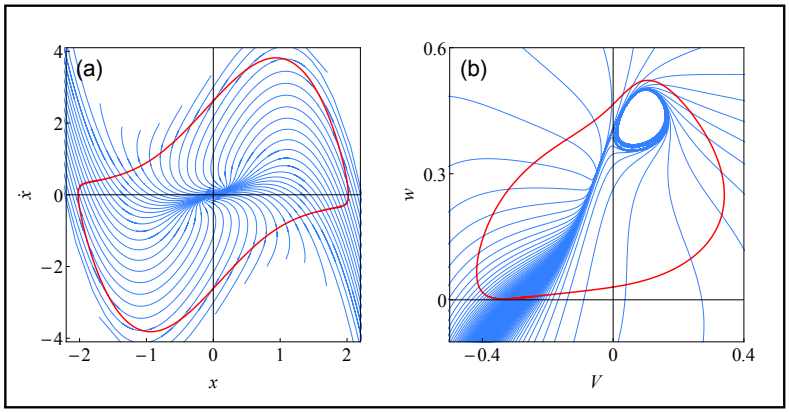


FIGURE S2.6: The isochrone structure of the two oscillators used in the main text, (a): van der Pol, Eq. (2.9) and (b): Morris-Lecar, Eq. (3.10). The limit cycle is plotted with a thick red line while the isochrones with thin blue ones. In (a) there are 50 isochrones corresponding to equal phase intervals, while in (b) there are 200.

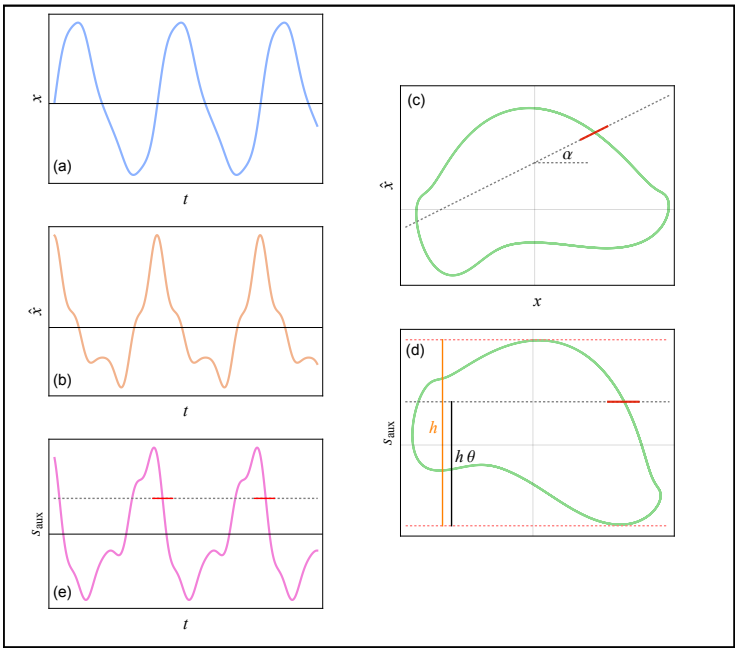


FIGURE S2.7: A schematic depiction of period determination with the use of signal embedding and inclined surfaces of section, as used for Fig. 2.6 in the main text. In (a) the measured signal $x(t)$ and in (b) the proxi variable $\hat{x}(t)$ (in our case we used the first derivative). In (c) the embedded signal (with green) as well as the chosen surface of section depicted with a straight red line. In (d) the embedded signal rotated by α so that the chosen surface of section becomes paralel to the horizontal axis. In (e) the vertical projection of the rotated embedded signal which is to be thresholded.

structure in the vicinity of the limit cycle, see Fig. S2.8. First, we embed our signl $x(t)$ in two dimensions (see Fig. S2.7 above). Then, considering the embedded signal in polar cooridnates, we average the radius variable over the angle variable with finite binning to obtain an approximation of the unpertubed limit cycle. On the obtained limit cycle approximation we choose points uniformly distributed in angle and for each point try several Poincaré sections at different angles. The angle corresponding to the lowest error should closely match the local isochrone.

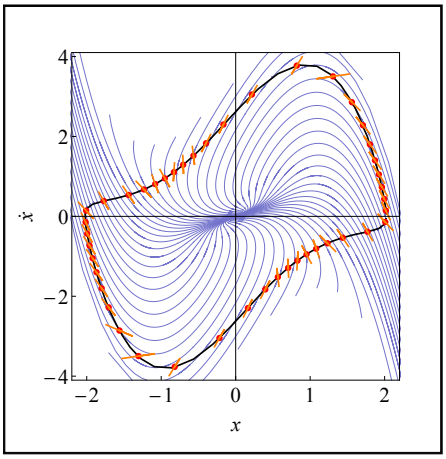


FIGURE S2.8: An estimated local isochrone structure using only one time series of length $t_{\text{sim}} = 500$ (coresponding to roughly 500 periods). The true isochrones are depicted with thin purple curves. The points where the inclination of the isochrones is estimated (in red) are uniformly distributed in angle. The estimated isochrones are depicted with orange lines. Parameters are $\varepsilon\|Z\| = 1$ and $\tau = 0.01$.

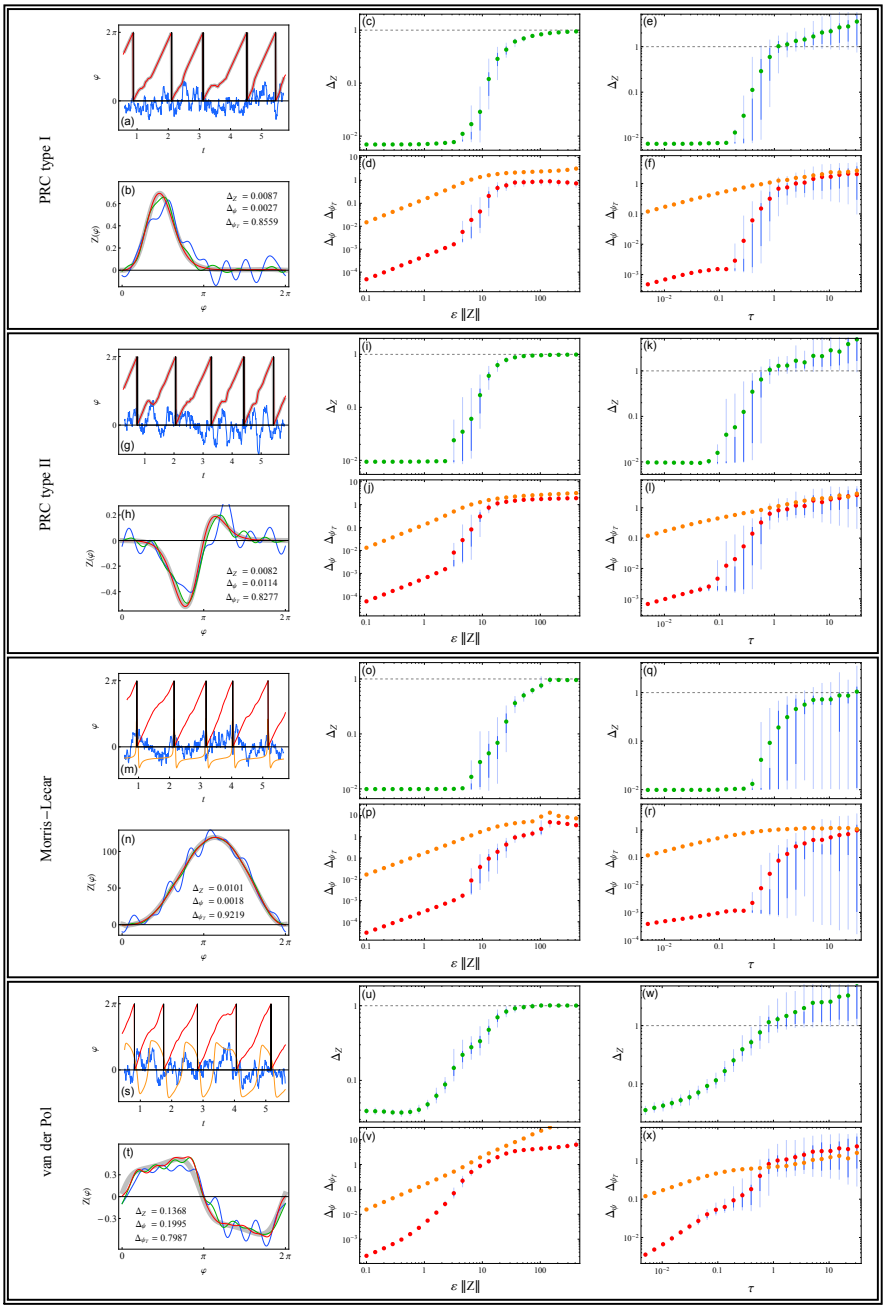


FIGURE S2.9: Overview comparison of test oscillators: PRC type I, PRC type II, Morris-Lecar and van der Pol. In plots (a, g, m, s) the signal, in (b, h, n, t) the PRC reconstruction, in (c, d, i, j, o, p, u, v) the effect of driving amplitude ε and in (e, f, k, l, q, r, w, x) the effect of driving correlation time τ. The parameters are $\varepsilon\|Z\| = 5$ and $\tau = 0.1$ in (a, b, g, h, m, n, s, t), $\tau = 0.1$ in (c, d, i, j, o, p, u, v) and $\varepsilon\|Z\| = 3$ in (e, f, k, l, q, r, w, x).

Chapter 3

Reconstructing networks

We present an approach for reconstructing networks of pulse-coupled neuron-like oscillators from passive observation of pulse trains of all nodes. It is assumed that units are described by their phase response curves and that their phases are instantaneously reset by incoming pulses. Using an iterative procedure, we recover the properties of all nodes, namely their phase response curves and natural frequencies, as well as strengths of all directed connections.

Adapted from: R. Cestnik and M. Rosenblum, Reconstructing networks of pulse-coupled oscillators from spike trains, *Physical Review E* 96, 012209 (2017) [51].
Associated software: Network inference @ Software section at the end.

3.1 Introduction

Reconstruction of a network structure from observations is an important problem relevant for many different areas such as neuroscience [74], physiology [75], climatology [76], genetics [77], ecology [78], etc. A group of established reconstruction techniques relies on analysis of the system's response to a specially designed perturbation, i.e. on invasive measurements [79]. However, often invasive measurement is not an option, e.g. in problems related to climatology, physiological studies, and medical diagnostics. In such cases one is restricted to analysis of observations of the free-running system.

Roughly speaking, there are two approaches to the problem. The first one does not imply any assumptions about the dynamics of the nodes and properties of the links and relies on different statistical and information-theoretical techniques for quantification of all connections [80]. In the second, model-based approach, some properties of the nodes (e.g. existence of a stable limit cycle) and of the links (e.g. weakness of coupling) are assumed to be known [81, 63, 82, 83]. In the present work we follow and extend the model-based approach. The main assumption is that the networks can be modeled by coupled limit cycle oscillators [15, 56]. In this way we follow our previous studies, where we have reconstructed the connectivity of a weakly coupled network of noisy limit-cycle or weakly chaotic oscillators for the case when the measurements allow for the determination of instantaneous phases [82, 84], see also [85, 86].

In this chapter we address the case when the signals are spiky, namely, that the measurements between the spiking events are dominated by noise and only determination of the times of spikes is reliable. Hence, the data we analyze are spike trains and estimation of time-continuous phase is not feasible. Next, we assume that effect of a chosen unit on the rest of the network is restricted to the time instant when the unit generates a spike. Thus, we use the model of pulse-coupled neuron-like oscillators [87, 88]. Assuming that the outputs of all nodes are known, that there are no unobserved external inputs, and that the coupling between the elements is sufficiently weak to justify the phase dynamics description, we recover the connectivity of the network and properties of all its nodes.

The chapter is organized as follows. In Section 3.2 we describe in details the model and summarize all the assumptions. In Section 3.3 we introduce our technique and in Section 3.4 we present the results of numerical studies. Section 3.5 presents discussion of the results.

3.2 The model

Our basic model for the network's node is a limit cycle oscillator which issues a spike when its phase φ achieves 2π. (We consider the phases wrapped to the $[0, 2\pi)$ interval, i.e. after the spike generation the phase of the unit is reset to zero). This spike affects all other units of the network according to the strength of the corresponding outcoming connections. Let the size of the network be N and let the connectivity be described by an $N \times N$ coupling matrix $\mathcal{E}$, whose elements ε_{ij} quantify the strength of the coupling from unit j to unit i. Between the spiking events, phases of all units obey $\dot{\varphi}_i = \omega_i$, where ω_i are frequencies. If unit i receives a spike from oscillator j, then it reacts to the stimulus according to its so-called phase response curve (PRC),

$Z_i(\varphi)$ [16, 55]. This means that the phase of the stimulated unit is instantaneously reset, $\varphi_i \to \varphi_i + \varepsilon_{ij} Z_i(\varphi_i)$.

Notice that oscillators are generally non-identical: they have different frequencies and different PRCs. However, we assume that response of the unit i to the stimuli from different units is described by the same PRC Z_i. Furthermore, we assume that PRCs are continuous. Next, the coupling is taken to be bidirectional but generally asymmetric, i.e. $\varepsilon_{ji} \neq \varepsilon_{ij}$, and there is no self-action, i.e. $\varepsilon_{ii} = 0$. Finally, we assume that there is no delay in coupling, i.e. that time of pulse propagation is much smaller than minimal inter-spike interval and can be neglected.

In neuronal modeling one commonly identifies two types of PRCs: if spikes always shorten the period of the stimulated unit, then the PRC is classified as type I. Otherwise, if depending on the phase of the stimulation, the period can be either shortened or prolonged, then the PRC is classified as type II [64, 55]. We model the type I PRC as

$$Z(\varphi) = (1 - \cos(\varphi)) \exp\left(3[\cos(\varphi - \varphi_0) - 1]\right) , \tag{3.1}$$

and the type II PRC as

$$Z(\varphi) = -\sin(\varphi) \exp\left(3[\cos(\varphi - \varphi_0) - 1]\right) , \tag{3.2}$$

where the parameter values are $\varphi_0 = \pi/3$ and $\varphi_0 = 0.9\pi$, respectively. The plots of these curves are shown in Fig. 3.1. Simulating this model, we generate N point

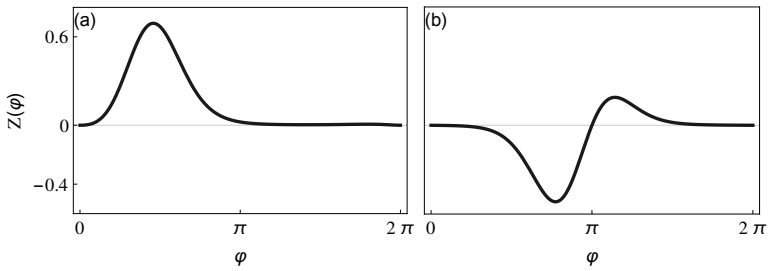


FIGURE 3.1: Model phase response curves of type I (a) and type II (b).

processes (spike trains) and then use them for network reconstruction, where we estimate the coupling matrix $\mathcal{E}$, PRCs $Z_i(\varphi)$, and frequencies ω_i of all elements, as discussed in the next section. There we also present the results of a test with a more realistic model of coupled Morris-Lecar neuronal oscillators [65].

3.3 The technique

For each node we reconstruct its properties as well as strength of all incoming connections. For definiteness, we always determine these quantities for the first node; the procedure then shall be repeated for all other units. Thus, we recover ε_{1j}, Z_1, and ω_1; for simplicity of presentation, in the following we omit the subscript 1.

We solve the reconstruction problem by iterations. First, since we do not have any *a priori* knowledge of the system, we assign some values to the coupling coefficients (we discuss several options of how this can be done) and use them in order to obtain a first estimate of the PRC. The knowledge of the latter allows for an improved

estimation of the network connectivity, which is then in turn used to obtain a better approximation of the PRC, and so on. We demonstrate that the procedure converges quite fast.

3.3.1 Notations and phase equations

Let the pulse train of the first oscillator contain $M+1$ spikes at times $t_k^{(1)}$, so that we have M inter-spike intervals $T_k = t_{k+1}^{(1)} - t_k^{(1)}$. In the following we treat each interval separately. Suppose that within the inter-spike interval T_k the first unit receives n stimuli from the unit i, we denote this number as $n_k(i)$. These stimuli appear at instants of time $t_k^{(i,l)}$, $l = 1, \ldots, n_k(i)$. The times relative to the beginning of the interval are denoted as $\tau_k^{(i,l)} = t_k^{(i,l)} - t_k^{(1)}$; see Fig. 3.2 for illustration. Respective

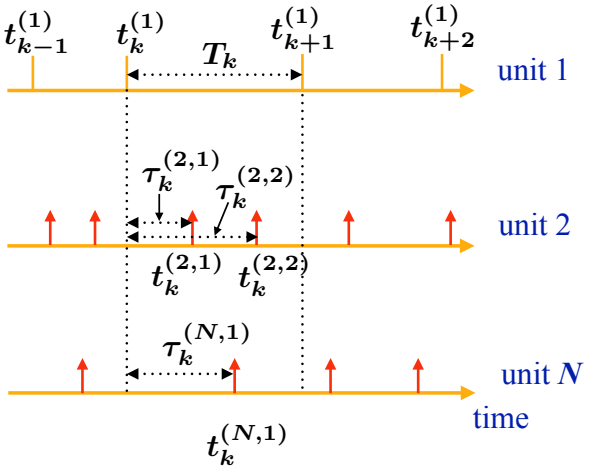


FIGURE 3.2: Illustration of notations used. T_k is an inter-spike interval of the driven unit. $\tau_k^{(i,l)}$ is the time (relative to the beginning of the interval T_k) when the spike number l from the unit number i arrives.

phases of the first unit are denoted as $\varphi(t_k^{(1)} + \tau_k^{(i,l)}) = \varphi_k^{(i,l)}$.

The phase increase within each inter-spike interval is

$$\omega T_k + \sum_{i=2}^{N} \varepsilon_i \sum_{l=1}^{n_k(i)} Z(\varphi_k^{(i,l)}) = 2\pi \,, \tag{3.3}$$

where the first term reflects the autonomous dynamics, whereas the second term describes the effect of pulse coupling. M inter-spike intervals yield a system of M Eqs. (3.3) for unknown coupling coefficients ε_i, frequency ω, and the PRC Z of the driven unit.

Assume for the moment that the coupling coefficients ε_i are given. Then, representing the unknown $Z(\varphi)$ as a finite Fourier series of order N_F, we obtain from Eqs. (3.3) a system of linear equations for $2N_F + 1$ Fourier coefficients and the unknown frequency ω. For a long time series, $M > 2N_F + 2$, this is an over-determined system which can be solved, e.g. by a least-mean-square fit or by singular value decomposition, see [89]. On the other hand, if PRC is given, we again obtain a solvable

linear system for $N-1$ coupling coefficients ε_i and frequency ω [1]. Thus, having an initial estimate for either PRC or coupling coefficients (practically we use the latter option) we can try to solve Eqs. (3.3) by iterations.

3.3.2 First iteration

The phases within each inter-spike interval vary from zero to 2π. For the first iteration we take the simplest approximation, i.e. we compute the phases as growing proportionally to time. Hence, when a spike at $\tau_k^{(i,l)}$ arrives, the phase of the first unit is taken as

$$\varphi_k^{(i,l)} \approx 2\pi\tau_k^{(i,l)}/T_k \,. \tag{3.4}$$

Since in this approximation we neglect the phase resets, $\varphi \to \varphi + \varepsilon_i Z(\varphi)$, the errors of such a phase estimation are of the order of $\varepsilon_i\|Z\|$, where $\|\cdot\|$ is the norm of the function, and accumulate with the number of the incoming spikes.

Next, we have to choose some initial values for the coupling coefficients ε_i. There are several options on how to do this. Firstly, we can exploit the simple idea that if there is no connection to the first unit from the unit i, then T_k cannot depend on the phase when the spikes from this unit appear, i.e. there shall be no dependence of T_k on $\varphi_k^{(i,1)}$. On the other hand, if this connection exists, the dependence of T_k on $\varphi_k^{(i,1)}$ shall be present as well; moreover, the larger ε_i, the stronger this dependence shall be. As shown in Appendix S3.1, this idea indeed works well for long time series. Alternatively, we can assign the same value to all ε_i, or we can take them randomly.

3.3.3 Next iterations

In the first approximation we compute the phases proportionally to time, see Eq. (3.4). If the coupling strength, ε_i, and parameters of the system, i.e. ω and Z, are already estimated, then we can use this knowledge for a more precise estimation of the phases. For illustration, suppose that within the inter-spike interval T_k the first unit receives three stimuli at times $\tau_k^{(i,1)} < \tau_k^{(m,1)} < \tau_k^{(n,1)}$. Then the phases at these three instances are computed as

$$\begin{aligned}
\varphi_k^{(i,1)} &= \omega\tau_k^{(i,1)} \,, \\
\varphi_k^{(m,1)} &= \omega\tau_k^{(m,1)} + \varepsilon_i Z\left(\varphi_k^{(i,1)}\right) \,, \\
\varphi_k^{(n,1)} &= \omega\tau_k^{(n,1)} + \varepsilon_i Z\left(\varphi_k^{(i,1)}\right) + \varepsilon_m Z\left(\varphi_k^{(m,1)}\right) \,.
\end{aligned}$$

The phase at the end of the given inter-spike interval is

$$\psi = \omega T_k + \varepsilon_i Z\left(\varphi_k^{(i,1)}\right) + \varepsilon_m Z\left(\varphi_k^{(m,1)}\right) + \varepsilon_n Z\left(\varphi_k^{(n,1)}\right) \,. \tag{3.5}$$

By definition, this value should be equal to 2π. However, since ω and Z are not exact, ψ generally differs from 2π. Therefore, we re-scale all phase estimates by the factor $2\pi/\psi$.

[1] When neither PRC nor ε_i are known, Eqs. (3.3) represent a nonlinear system with respect to $N+2N_F+2$ unknowns. Alternatively, one could consider products of ε_i and the Fourier coefficients as unknowns and end up with a linear, but rather large system of $N(2N_F+2)$ unknowns.

Now, using the newly estimated phases and the estimation of the PRC from the previous iteration, we can compute new values of the coupling coefficients ε_i, and then repeat the whole procedure. As we demonstrate below, these iterations converge quite quickly.

3.4 Numerical tests

In this section we present the results of numerical testing of our reconstruction algorithm. For this goal we generate the networks with some randomly chosen parameters and simulate them to obtain the data. Next, we recover the network from these data and compare the reconstructed parameter values with the true ones.

3.4.1 Basic model

In this test, for acceleration of data generation we assign identical PRCs to all nodes; however, because each node of the network is treated separately, this feature is neither used for reconstruction nor implies any limitations in the applicability of the algorithm. First, we consider networks of $N = 20$ oscillators, with natural frequencies taken from a uniform distribution between 1 and 2. Strength of network links is sampled from the positive half of a Gaussian distribution with zero mean and standard deviation 0.02. The frequency of the first oscillator is set to 1, assuring that it is the slowest one (as discussed below, this is the most difficult case) and then we reconstruct its PRC Z, frequency ω, and strength of all incoming links ε_i, $i = 2, \ldots, 20$. We exclude from the consideration the networks where at least two units synchronized. We use ten iterations of the procedure described above. The number of the Fourier harmonics approximating the PRC is $N_F = 10$ which is sufficient to describe PRCs of common neuronal models.

Before presenting the results we recall that all equations contain ε_i and Z in products of each other. Therefore, solutions $\{\varepsilon_i, Z\}$ and $\{c\varepsilon_i, Z/c\}$, where c is an arbitrary constant, are equivalent. The factor c has no physical meaning by itself, but since we want to compare the reconstructed values with the originally given, we have to fix it. Quite arbitrarily, we do it by minimizing

$$\sum_{i=2}^{N} \left[\varepsilon_i^{(\mathrm{t})} - c\varepsilon_i^{(\mathrm{r})}\right]^2$$

where the superscripts (t) and (r) stand for true and reconstructed, respectively. This condition yields

$$c = \sum_{i=2}^{N} \varepsilon_i^{(\mathrm{t})}\varepsilon_i^{(\mathrm{r})} \Big/ \sum_{i=2}^{N} \left[\varepsilon_i^{(\mathrm{t})}\right]^2 . \tag{3.6}$$

Using this normalization, we show the results of a particular run in Fig. 3.3; for this computation we took initially $\varepsilon_i = 1\,,\ \forall i$.

Next, we perform a statistical analysis for 10^5 network configurations. To quantify the quality of the reconstruction, we define the corresponding errors for recovered PRC, ε_i and ω as

$$\Delta^2_{\mathrm{PRC}} = \frac{\int_0^{2\pi} \left[Z^{(\mathrm{t})}(\varphi) - Z^{(\mathrm{r})}(\varphi)\right]^2 d\varphi}{\int_0^{2\pi} \left[Z^{(\mathrm{t})}(\varphi)\right]^2 d\varphi}\,, \tag{3.7}$$

$$\Delta_\varepsilon^2 = \sum_{i=2}^{N} \left[\varepsilon_i^{(\mathrm{t})} - \varepsilon_i^{(\mathrm{r})}\right]^2 \Big/ \sum_{i=2}^{N} \left[\varepsilon_i^{(\mathrm{t})}\right]^2 , \tag{3.8}$$

and

$$\Delta_\omega^2 = \left[\omega^{(\mathrm{t})} - \omega^{(\mathrm{r})}\right]^2 , \tag{3.9}$$

respectively [2]. The distributions of errors, shown in Fig. 3.4, confirm robustness of the iterative procedure.

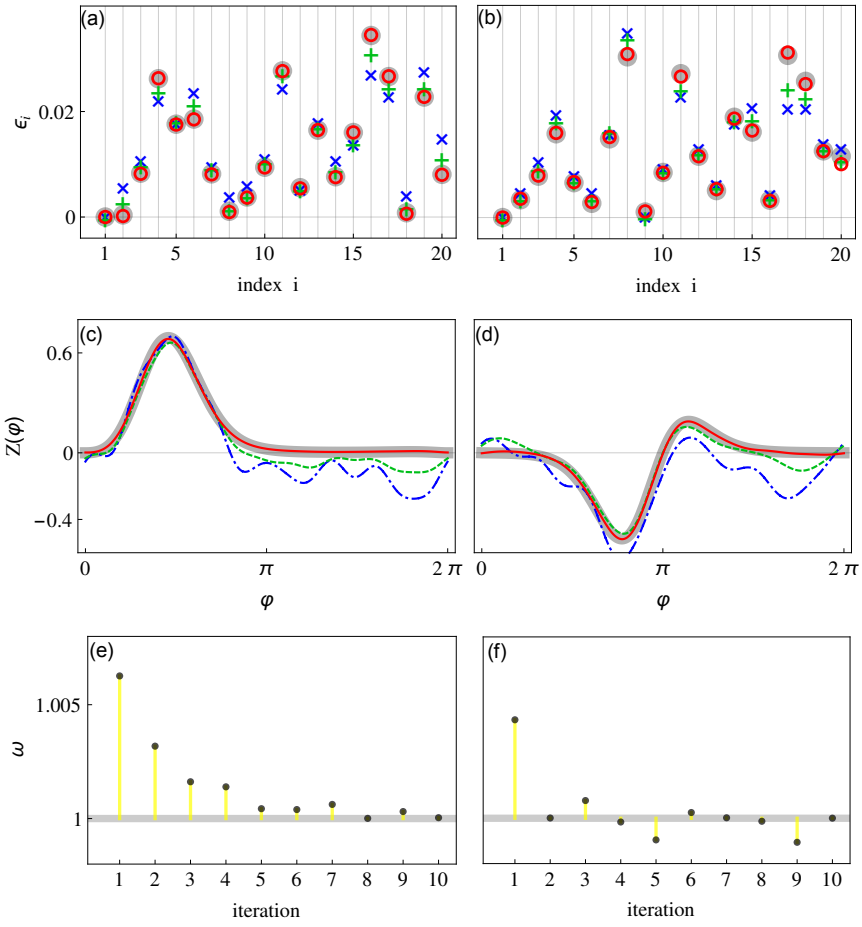


FIGURE 3.3: Reconstruction of a network of 20 units using $M = 200$ inter-spike intervals. Panels (a,c,e) and (b,d,f) show the results for PRC type I and PRC type II, respectively, see Eqs. (3.1,3.2). (a,b) Strength of incoming connections for the first oscillator: true values (gray disks) and values recovered after one, 2, and 10 iterations (blue crosses, green pluses, and red circles, respectively). (c,d) True (wide gray curve) and reconstructed PRCs, after one, 2, and 10 iterations (blue dashed-dotted, green dashed, and red solid curves, respectively). (e,f) Estimated natural frequencies as functions of the iteration number; the true value $\omega = 1$ is shown by horizontal gray line. Notice that in this figure we illustrate the most difficult case of the slowest node.

[2] Notice that error Δ_ε is minimized due to normalization according to Eq. (3.6).

Figure 3.5 shows the dependence of the reconstruction error on the number of inter-spike intervals. Naturally, the more data we use, the better results we expect. This test demonstrates, that reasonable reconstruction can be achieved already for several hundreds of intervals. Additionally, instead of initially assigning the same value to all coupling coeffiecient ε_i, we also performed the test with random assignment of the initial values. For several generated networks we performed 10^4 reconstructions with different initial ε_i. The results confirm convergence of the algorithm for this case as well.

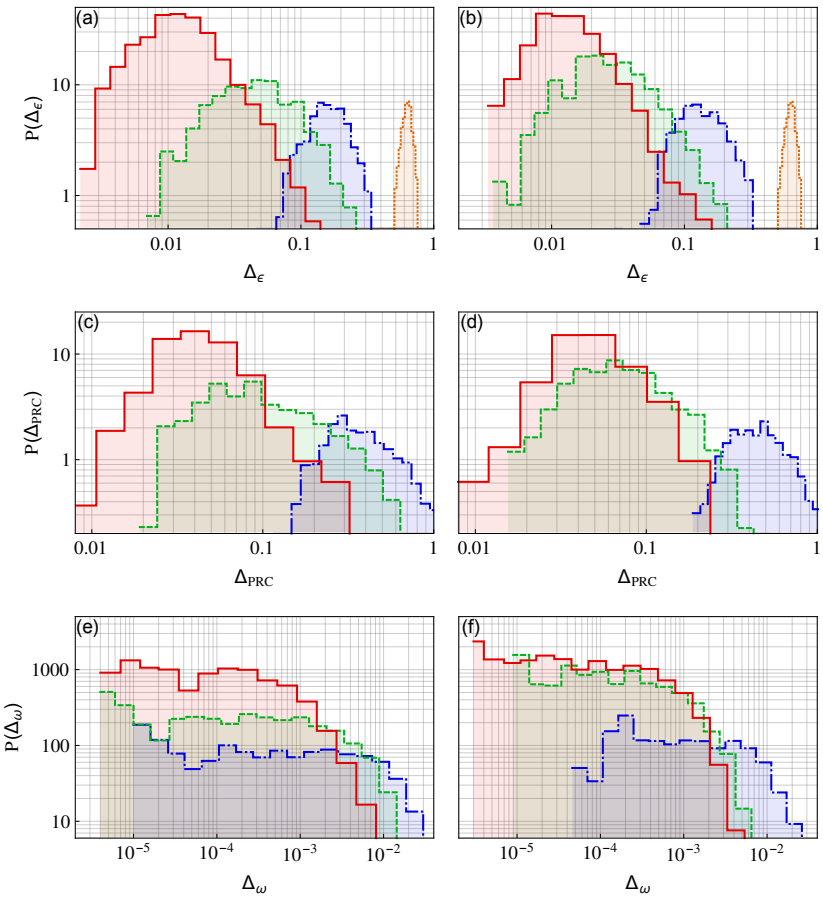


FIGURE 3.4: Histograms of errors of the coupling strengths Δ_ε (a,b), PRC Δ_{PRC} (c,d), and frequency ω (e,f), see Eqs. (3.7-3.9). Panels (a,c,e) and (b,d,f) correspond to tests with PRC type I and type II, respectively. In each panel the results of the first, third, and tenth iterations are shown in blue (dash-dotted), green (dashed), and red (solid line) respectively. In (a,b) also the distribution of errors for initial values $\varepsilon_i = 1$ is shown in orange (dotted line).

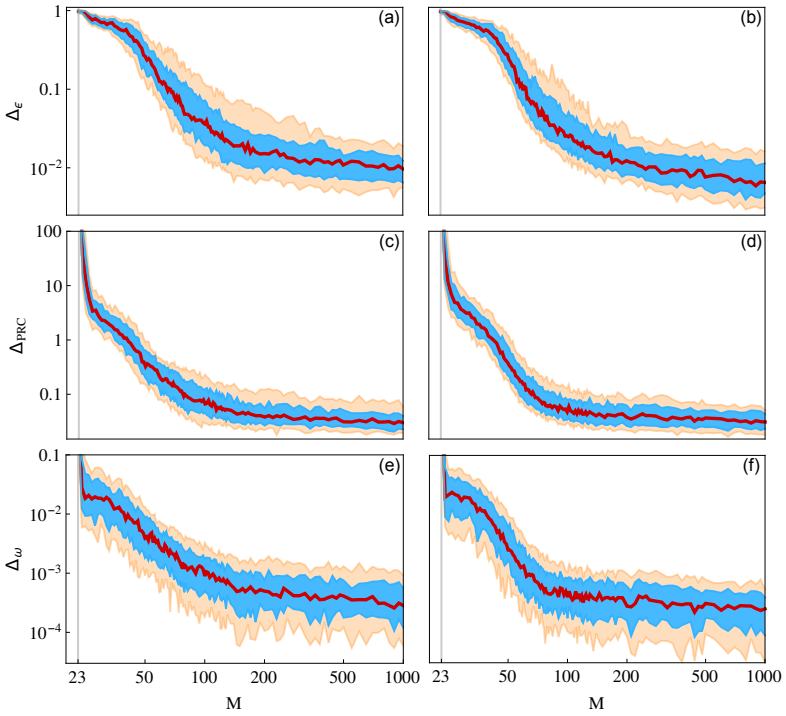


FIGURE 3.5: Error of the reconstruction in dependence on the number of inter-spike intervals M used for the analysis. Panels (a,c,e) and (b,d,f) show the results for PRC type I and PRC type II, respectively. Panels (a,b) show the error of the coupling coefficients ε_i (3.7), panels (c,d) the error of the PRC (3.8) and panels (e,f) the error of the frequency ω (3.9). For each value of M reconstruction error was computed for 6000 different networks. The blue and the orange areas contain 50% and 75% of the errors, respectively; the median is shown in red.

To conclude the presentation of the algorithm we briefly discuss its stability with respect to noise as well as its computational complexity. We have tested our method on data generated with dynamical white noise, $\dot{\varphi}_i = \omega_i + \xi_i$, and found that for weak noise, the errors of the reconstruction scale roughly linearly with the standard deviation of the noise, σ, i.e. $\Delta_X = \kappa_X \sigma$, where scaling coefficients are $\kappa_\varepsilon \approx 40$, $\kappa_{\mathrm{PRC}} \approx 100$ and $\kappa_\omega \approx 1$. Regarding the computational complexity of the algorithm, we note that, at each iteration, recovering incoming connections of one node requires least squares minimization of an $M \times N$ system (for ε_i) and $M \times (2N_F + 2)$ system (for PRC). Assuming that N_F can be kept constant, we arrive at $\mathcal{O}(N^3 M)$ asymptotic scaling dependence for reconstruction of the whole network. Since the number of intervals M should always be greater than the number of oscillators N in order to have an over-determined system, we took $M \propto N$ and measured the computation time for different system sizes ranging from $N = 10$ to $N = 500$. Within that range the error of the reconstruction was roughly constant and the computation time scaled like $\mathcal{O}(N^4)$, as expected. In fact, computations are very fast: for $M = 1000$ inter-spike intervals, a single iteration of the reconstruction of one node's incoming connections in a network of 100 oscillators takes about a second on a common computer.

3.4.2 Network of Morris-Lecar neurons

In the next test we make a step towards more realistic modeling and consider a network of Morris-Lecar neurons [65]. The equations of the network are:

$$\begin{aligned}\dot{V}_i =& I_i - g_L(V_i - V_l) - g_K w_i (V_i - V_K) \\ & - g_{Ca} m_\infty(V_i)(V_{Ca} - V_i) + I_i^{(\mathrm{syn})} \,, \\ \dot{w}_i =& \lambda(V_i)(w_\infty(V_i) - w_i) \,,\end{aligned} \tag{3.10}$$

where

$$\begin{aligned} m_\infty(V) &= [1 + \tanh(V - V_1/V_2)]/2 \,, \\ w_\infty(V) &= [1 + \tanh(V - V_3/V_4)]/2 \,, \\ \lambda(V) &= \cosh[(V - V_3)/(2V_4)]/3 \,, \end{aligned} \tag{3.11}$$

and $I_i^{(\mathrm{syn})}$ is the total incoming synaptic current. We write the latter as

$$I_i^{(\mathrm{syn})} = [V_{\mathrm{rev}} - V_i] \sum_{j, j\neq i} \frac{\varepsilon_{ij}}{1 + \exp[-(V_j - V_{\mathrm{th}})/\sigma]} \,. \tag{3.12}$$

We take standard values for most of the parameters [3]. Parameters of the synaptic coupling are $V_{\mathrm{rev}} = 0.2$, $V_{\mathrm{th}} = 0.25$, and $\sigma = 0.01$. The neurons are non-identical: the values of the current are $I_i = 0.077\big[1 + 0.22\,\mathcal{U}(0,1)\big]$, where $\mathcal{U}(0,1)$ is a uniformly distributed random number between zero and one; for these values the neurons remain in the spiking state. Similarly to the previous test, we assure that the first oscillator is the slowest by setting its current to the minimal value, $I_1 = 0.077$. The results of the analysis for 20 neurons and 200 inter-spike intervals, shown in Fig. 3.6, confirm the efficiency of our technique.

3.5 Discussion and conclusions

With the help of two model systems we have demonstrated, that our technique provides a robust reconstruction of a network. The data requirements are not too demanding: the reconstruction is quite precise already for time series of several hundreds of spikes. Now we discuss some limitations of the method.

Our approach is based on modeling phase dynamics of a network from data, which assumes weak coupling approximation. If the coupling is not weak enough, the description in terms of PRCs can still be reasonable, but in this case the phase response can be also amplitude-dependent. This means that estimation of the network connectivity remains correct, but the obtained PRC has a limited applicability. Unfortunately, the assumption of weak coupling can be hardly verified from data analysis only. However, we can check the quality of our phase dynamics modeling by looking at the deviation of the phase ψ, see Eq. (3.5), from 2π. Furthermore, we suggest to perform the reconstruction of the same network many times, starting from different initial values of the coupling strength ε_i and looking for the convergence of the results: if different initial values yield close network parameters, then

[3] Parameters of the system (3.10,3.11) are: $g_L = 0.5$, $g_K = 2$, $V_1 = -0.01$, $V_2 = 0.15$, $V_{Ca} = 1$, $V_K = -0.7$, $V_L = -0.5$, $g_{Ca} = 1.33$, $V_3 = 0.1$, $V_4 = 0.145$.

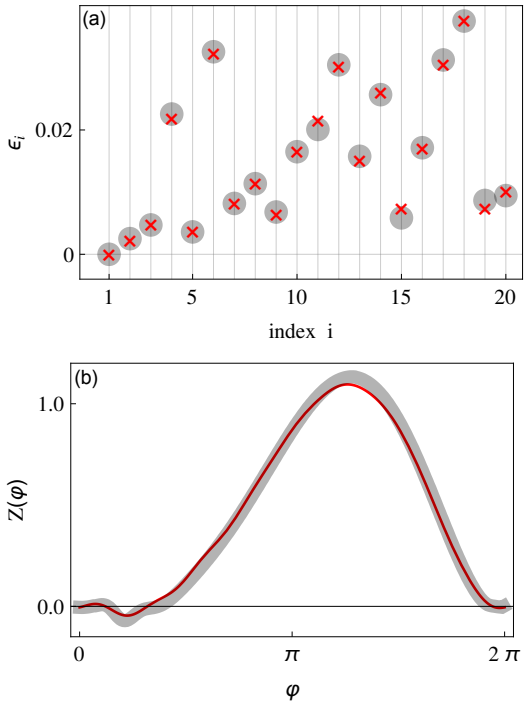


FIGURE 3.6: Reconstruction of a network of 20 Morris-Lecar neuronal oscillators, see Eqs. (3.10-3.12). 200 inter-spike intervals were used for the reconstruction. (a) True values of the strength of the incoming connections for the first neuron (gray disks) and these values reconstructed after 10 iterations (red crosses). (b) True (wide gray curve) and reconstructed (solid red curve) PRC. Notice that in this figure we illustrate the most difficult case of the slowest node.

the results can be trusted. Now we comment on the initial estimate of phases using Eq. (3.4). As already mentioned, the error is proportional to $\varepsilon_i \|Z\|$ and increases with the number of spikes that arrive within the inter-spike interval of the driven unit. This explains why the case of the slowest oscillator is the most difficult one: such an oscillator has on average more incoming stimuli per inter-spike interval then the fast units. This means, that although our examples demonstrate robust reconstruction, it may fail if $\omega_i/\omega_1 \gg 1$.

Next limitation is related to variability of the inter-spike intervals of the driving unit i (for $\omega_i > \omega_1$). Indeed, suppose that drive is strictly periodic. Then time of the appearance of the first spike unambiguously determines the timing of the following ones, and hence, the length of the inter-spike interval T_k. However, T_k is then determined by the sum of different pieces of PRC and this sum cannot be disentangled. The initial estimation of the strength of the connection as described in Appendix S3.1 can still work, but the recovery of the PRC becomes impossible and the iterative procedure fails. So, we foresee that reconstruction may be not so robust for very sparse networks where we expect to have purely periodic nodes. On the other hand, a realistic network is noisy, and noise naturally provides the desired variability in the time series, thus enhancing the reconstruction. Finally, we mention that the reconstruction fails if the network synchronizes.

Acknowledgment

We acknowledge useful discussions with A. Pikovsky, M. Zochowski, R. Andrzejak, and A. Daffertshofer. This work has been financially supported by the EU project COSMOS (642563). Numerical calculations of M. Rosenblum were supported by the Russian Science Foundation Grant No. 17-12-01534.

S3.1 First estimation of incoming connections

For sufficiently long data an initial estimation of the coupling strength ε_i can be performed by evaluating the effect of the first pulse from unit i, that arrives within the k-th inter-spike interval, on the length of this interval T_k. For this purpose, we first plot T_k vs $\varphi_k^{(i,1)}$, for all incoming links. Next, for each plot, we divide the φ-axis into N_b bins and average the T_k values within each bin. As a result, we obtain a dependence $\bar{T}_n^{(i)}(\bar{\varphi}_n)$, where $\bar{\varphi}_n = \frac{\pi}{N_b}(2n-1)$, $n = 1, \ldots, N_b$, are phases at the centers of bins. Our conjecture is that $\bar{T}_n^{(i)}(\bar{\varphi}_n)$ reflects the strength of the incoming connection: if this strength is zero, i.e. there is no incoming link from unit i, then there shall be no dependence; on the other hand, if the incoming connection is strong, then we expect the dependence to be well-pronounced.

We illustrate this idea in Fig. S3.1, where two such plots are shown for the cases of strong and weak incoming connections.

We see that indeed $\bar{T}_n^{(i)}(\bar{\varphi}_n)$ reflects the coupling coefficients ε_i. Hence, we use the standard deviation of this dependence as the first estimate, i.e. we take

$$\varepsilon_i = \left\langle \left(\bar{T}_n^{(i)} - \langle \bar{T}_n^{(i)} \rangle\right)^2 \right\rangle^{1/2}, \tag{S13}$$

where $\langle \cdot \rangle$ means averaging over N_b bins. Although the correspondence between the estimate and true coupling strength is not exact, in most cases this approach yields reasonable values.

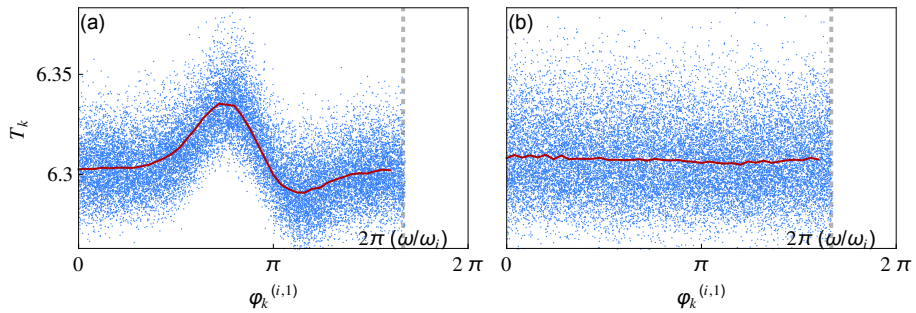


FIGURE S3.1: Scatter plots of inter-spike intervals T_k vs. approximated phase $\varphi_k^{(i,1)}$ of the first spike from a chosen driving oscillator i, for a strong coupling strength (a) and for a weak one (b). The horizontal axis is divided into $N_b = 50$ bins and solid red curve shows the average of T_k over each of the bins, $\bar{T}^{(i)}$, as a function of the central phase $\bar{\varphi}$ of the bins; $2 \cdot 10^4$ inter-spike intervals are used in this computation.

Chapter 4

Inferring the dynamics using recurrent neural networks

We investigate the predictive power of recurrent neural networks for oscillatory systems not only on the attractor, but in its vicinity as well. For this we consider systems perturbed by an external force. This allows us to not merely predict the time evolution of the system, but also study its dynamical properties, such as bifurcations, dynamical response curves, characteristic exponents etc. It is shown that they can be effectively estimated even in some regions of the state space where no input data were given. We consider several different oscillatory examples, including self-sustained, excitatory, time-delay and chaotic systems. Furthermore, with a statistical analysis we assess the amount of training data required for effective inference for two common recurrent neural network cells, the long short-term memory and the gated recurrent unit.

Adapted from: R. Cestnik and M. Abel, Inferring the dynamics of oscillatory systems using recurrent neural networks, *Chaos* 29, 063128 (2019) [52].
Associated software: OscillatorSnap @ Software section at the end.

Inference of most dynamical properties of any system is typically best done with an active experiment, meaning that one has the power to repeatedly manipulate the system state in controlled conditions in order to isolate the desired measure, e.g. car technical inspection, where the examiners manipulate the car into specific situations in order to asses its safety on the road. Often however, one only has access to passive observations. This can be due to a number of reasons, for example, the system can be very big, like when studying the dynamics of our planet, or the system can be delicate, like when studying the physiology of the human body. The algorithms that attempt to distil dynamical measures from passive observations commonly make assumptions on how the observations were collected and typically require very long observations. Imagine, for example, assessing the car's capabilities only from observing routine trips to work. Here we propose a conceptually simple scheme, relying on the now well established artificial neural networks. In particular, we use recurrent neural networks which have established themselves in timeseries forecasting, text generation etc. and train the networks to mimic the system dynamics, allowing us to then perform an active experiment on the trained model. We test this on several oscillatory systems and measure their characteristic properties, such as, bifurcations, dynamical response curves, characteristic exponents and compare them to the measures from the original system.

4.1 Introduction

Oscillatory systems can be found in all fields of natural science: in optics [90], electronics [91], chemistry [92], biology [16], climatology, life science, etc. Oscillations are present at all scales, both temporal and spatial, e.g. in biology, from cells like neurons [93], to organs like the heart [94], to oscillations spanning the entire organism such as the circadian rhythm [95] and the menstrual cycle.

Classical modeling of dynamical systems consists of reasoning of the terms involved followed by directly assessing the validity of the model. For low-dimensional systems, this works well; however the aforementioned examples are all complex, high-dimensional and coupled to their surrounding, like the brain which consists of many coupled neurons [96]. Other examples include climate models [97] and fluid dynamics, which have been a major driving force for the investigation of periodic motion, synchronization of oscillatory systems [15], period doubling bifurcations, and chaotic oscillations. Accurate modeling of such systems is hard, but with increasing computer power existing methods to infer dynamical systems from measurements are easier to realize [98, 99, 100].

For high-dimensional systems, one has to either measure with many channels or apply embedding methods [101, 25], or commonly both. If a system is truly periodic, then it lives on a one-dimensional manifold and may be in principle modeled by a two-dimensional system of equations. If weakly perturbed, under certain assumptions [102], the system remains close to the unperturbed orbit. Such perturbations may originate from another oscillator, a network of oscillators or elsewhere from the environment. However, if the system is close to a bifurcation, perturbations may cause it to undergo dramatic changes in its dynamics. Bifurcations however, are hard to predict for heuristic models, whereas this is generally easier if equations are known. Under this point of view previous approaches using symbolic regression methods [99, 103, 104] proved successful. Heuristic methods such

as liquid state machines, echo state networks, or various types of artificial neural networks [105, 106, 107, 108, 43] perform very well in predicting dynamical systems. However, few studies are known for particular aspects of oscillatory systems inferred from time series. Here, we investigate several oscillatory models under perturbation, as they may occur in real measurements. Our focus is on the inference of dynamical properties, bifurcation behavior and chaos, even if not all of the parameter variation is included in the measurement.

In all of the above methods one typically *a priori* assumes a model (or a class of models), sets an optimization criterion (e.g. least squares) and optimizes model parameters or its functional constituents for nonparametric methods. Mathematical aspects are most often left aside, e.g. basic assumptions on the existence of solutions and robustness under perturbation, in particular for heuristic methods. Here we utilize the widely used artificial neural networks (ANNs). An ANN has several hyperparameters such as the actual topology of the network, the activation function, the learning rate, and is in general very pliable toward many different tasks. Since we consider time series, we investigate the capacities of recurrent neural networks [109] (RNN). Due to loops in their connectivity they retain past information, i.e. they inherently possess memory, similar to embedding. They tend to be particularly successful in speech recognition [110], text generation [49] and machine translation [48], where a forward-oriented semantic is present. The aim of this study is to evaluate how suited RNNs are for modeling oscillatory systems under the aspect of parameter change and perturbations. In this way, the inferred model of the oscillator can be probed via changing the perturbation signal, effectively allowing the performance of an active experiment.

The article is structured as follows: in section 4.1.1 we refer to relevant related works and briefly recall the RNN functioning. in section 4.2.1 we introduce the dynamical inference setup and training scheme. We then present numerical tests with example systems in section 4.2.2 where we compare signal reconstruction and other observables such as the phase response curve [55] (PRC) and the maximum Lyapunov exponent [111]. The different example systems are chosen as representatives of different mechanisms giving rise to oscillatory behavior; specifically, self-sustained and excitable oscillations, time-delay induced oscillations and chaos. We continue by presenting numerical tests on data requirement for successful inference in section 4.2.3 where we compare the inference quality for different lengths of time-series used for training. We present the methods used in more detail in section 4.3 and finally, discuss the novelties, limitations and generalizations of our approach in the discussion section 4.4.

4.1.1 Previous work

In this paragraph we chronologically go through works related to this chapter. In Ref. [112] the author uses a RNN for learning state space trajectories. In Ref. [113] the authors show that any trajectory generated by a finite-dimensional dynamical system can be effectively represented with a neural network. In Ref. [114] the authors model a dynamical system with a perturbation using a RNN. In Ref. [115] the authors use feed-forward neural networks [47] to model dynamical systems. They feed in delayed values of one variable as well as a control parameter as inputs and train the network for one step predictions. The approach works well and they reproduce bifurcation diagrams of several example dynamical systems. In our approach we train RNNs

for one step prediction where the input consists of several time-delayed values of one or more variables as well as an arbitrary number of perturbative signals (we will refer to the perturbative signal inputs as p-inputs). The past values of one or more variables contain information of the topology of the attractor of the complete system according to the Takens' delay embedding theorem [25]. The RNN topology prioritizes more recent values over older ones for the next prediction, therefore, we believe it is more suitable for time-series prediction and demonstrate its efficiency throughout this chapter.

4.1.2 Recurrent neural networks

Artificial neural networks is nowadays a relatively broad term as many different network topology classes are commonly used for dealing with different types of problems. The simplest class of ANNs are the feed-forward networks, they have directed connections between subsequent layers without any loops, effectively allowing the information to flow in only one direction - forward. The slightly more general class are the recurrent neural networks (RNN), they can have loops in their connectivity, which can result in internal state memory. Different RNNs then differ in the fine architecture of the basic cells, the order and type of logical operations. In this work we apply two commonly used cells, the long short-term memory cell [116] (LSTM) and the gated recurrent unit [117] (GRU). LSTM was constructed first in an attempt to deal with long term dependencies and GRU emerged as its faster simplification. Further details on the functioning of different RNN cells can be found in Ref [118]. The software implementation was accomplished with the help of TensorFlow [119] and Keras [120].

4.2 Results

4.2.1 Inference scheme

Consider a general dynamical system $\dot{\vec{x}}(t) = f(\vec{x}) \in \mathbb{R}^N$, perturbed by an external perturbation $\vec{p}(t) \in \mathbb{R}^N$. Suppose we have measured the timeseries of $n_x \geq 1$ state variables $\vec{x} = (x_1, x_2, \ldots, x_{n_x})$ as well as the n_p timeseries of the perturbation $\vec{p} = (p_1, p_2, p_3, ...)$ over a period of time. The question we investigate is if it is possible to recover both the autonomous dynamics of the system $\vec{x}$ and the system's response to the perturbation using RNNs. Without perturbation we can only recover the dynamics on the attractor, but with a perturbation the phase space around the attractor is explored and we have a means to infer the neighboring phase space, too.

We train the RNN to receive historical values of $\vec{x}(t), \vec{p}(t)$ and return the time-evolved state $\vec{x}(t + \Delta t)$. In practice, this is accomplished by first "unrolling" the network. The RNN at each time step can be represented as a separate copy of the same network, where the recurrent connections have been replaced with regular connections linking every copy with its successor. Then this chain of networks is truncated and a finite number of "rolls" (network copies) considered. The historical values effectively correspond to a time-delay embedding, allowing the RNN to infer the state of the system. The number of rolls therefore corresponds to the dimensionality of the time-delay embedding, although the time steps we use are typically much smaller then delays used in embeddings, making the succesive steps considerably correlated. Nevertheless we can deduce from the Takens' embedding theorem [25] that

at least $R > 2M + 1$ rolls have to be considered, where M is the dimensionality of the attractor. The sampling Δt must be smaller than the smallest time scale which occurs in the system (or which we may want to include in our modeling). Heuristically, one can say that the time-resolution Δt should be chosen fine enough to see the details of interest.

Given an appropriate resolution and number of network copies we can begin to "train" our model, i.e. to start a loop for the statistical inference method: At each training step the network "learns" the possible relation:

$$\vec{x}(t), \vec{p}(t) \mapsto \vec{x}(t + \Delta t) \tag{4.1}$$

using the time instants t, $t - \Delta t$, $t - 2\Delta t, ...,$ $t - (R - 1)\Delta t$. We use a least-squares optimization criterion $\frac{\|\vec{x}_e - \vec{x}\|}{\|\vec{x} - \langle\vec{x}\rangle\|}$ (where $\langle\cdot\rangle$ stands for the mean) to determine quantitatively how well the estimates $\vec{x}_e$ match the true values $\vec{x}$. Hereafter, we use estimated and modeled as synonymous.

4.2.2 Examples

We put our scheme to the test on several model systems, including time-delay, excitable and chaotic oscillators. The *validation test* consists of comparing the modeled signal with the original when presented with data never seen in training. As important measures of oscillatory systems we estimate the phase response curve [55] (PRC) and the maximal Lyapunov exponent [111] for comparing the predictive power the model has in a dynamical systems context.

For all examples shown in this chapter we use a network with 1 hidden layer of 32 nodes and 36 rolls. We use *tanh* activation for all but the output layer, where we use *linear* activation so that a continuous signal can be produced. There are two common cells used in RNN: the long short-term memory cell [116] (LSTM) and the gated recurrent unit [117] (GRU). We tested both for the systems in this study; as a result we found that GRU performed poorly, hence all results shown are for LSTM models, cf. Sec. 4.2.3 for a comparison of the two cells. To generate the data we first simulate the perturbation signal using the stochastic Euler-Maruyama integration scheme, and then integrate the dynamical equations with fourth order Runge Kutta. We use a sufficiently small time step and then re-sample the signals to an appropriately lower time resolution to create the network training data. The resolution is chosen such that 36 points (the number of considered historical values R) corresponds to 1 natural period of the oscillator. In the case of chaotic oscillators, this was computed as the average period, in the case of excitable systems the time needed to return from the excited state to the fixed point was used.

Roessler oscillator - phase response curve, bifurcation diagram and Lyapunov exponents

For our first test we use the Roessler system [4], because it exhibits many different regimes, i.e. simple periodic oscillations, higher period oscillations and chaos, by varying just one parameter b, cf. Fig. 4.2(a) for the bifurcation diagram. The

corresponding equations, including the perturbation p read:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x - c) + p(t) \end{aligned} \tag{4.2}$$

with parameters $a = 0.2$ and $c = 5.7$. To explore the phase space, we can vary b through a constant term in the perturbation $p(t)$. For the first test we set $b = 2.0$, such that the system has a simple attractive periodic orbit. In the following we use a stochastic perturbation, generated by an Ornstein-Uhlenbeck [26] process:

$$\dot{q} = -q/\tau + \epsilon\sqrt{2/\tau}\xi(t) \tag{4.3}$$

where ξ is Gaussian white noise $\langle \xi(t)\xi(t')\rangle = \delta(t - t')$, and $\epsilon = 0.5$ and $\tau = 5.0$ are the amplitude and correlation time of q. The stochastic differential equation is integrated with the Euler-Maruyama method to obtain noise with exponentially decaying correlation: $\langle q(t)q(t')\rangle = \epsilon^2 e^{-(t-t')/\tau}$.

Now, we set $p(t) = q(t)$ and feed both the signal $x(t)$ and the perturbation signal into the network as described in section 4.2.1. The timeseries length corresponds to 1000 natural periods, which is presented to the network during 500 training epoch in batches of 100 time points with resolution $\Delta t = 0.17$ (the time step used for the integration is significantly smaller). The network is trained using stochastic gradient descent [121] with learning rate 0.005.

The network learns to reproduce the dynamics to a mean deviation of 2.5×10^{-2} (for the time window in Fig. 4.1), such that the reproduced signal is visually indistinguishable from the one generated with Eqs. (4.2). This holds true for both the perturbed signal - where $p(t)$ is fed to both the network and the equations, as well as for the unperturbed signal - where $p(t) = 0$ is used.

Can we use the inferred network for more than just mimicking a signal, e.g. to study dynamical regimes? We want to study this scenario in probing the network for dynamical responses to stimuli. Since the system in question is a self-sustained oscillator it is natural to estimate its PRC, cf. Sec. 4.3.2. The comparison of the estimate obtained from the RNN and the the true PRC, is displayed in Fig. 4.1. The coincidence is very good, up to mean deviation of 0.1 in the entire phase range $[0, 2\pi)$. Indeed, this can be an effective method of inferring the PRC from data, cf. [60, 58, 50].

We perform another test with the Roessler oscillator, this time testing the power of the network to reproduce the system across several dynamical regimes. For this we use $b = 0$ and a strong and varied p-input that considerably explores the state space:

$$p(t) = \frac{1}{2}\exp(q(t)) \tag{4.4}$$

where $q(t)$ is the p-input described in Eq. (4.3). It yields a process with a log-normal distribution: $P(p) \sim \frac{1}{p}\exp\left(-2\big(\log(p)+\log(2)\big)^2\right)$. Such p-input spans a wide range of values, effectively introducing different regimes of our system to the network, see Fig. 4.2(a) for the p-input probability distribution with respect to b bifurcation (grey shaded region in the background). The idea is that the network then effectively learns to mimic the regimes corresponding to different values of b, which we can invoke via

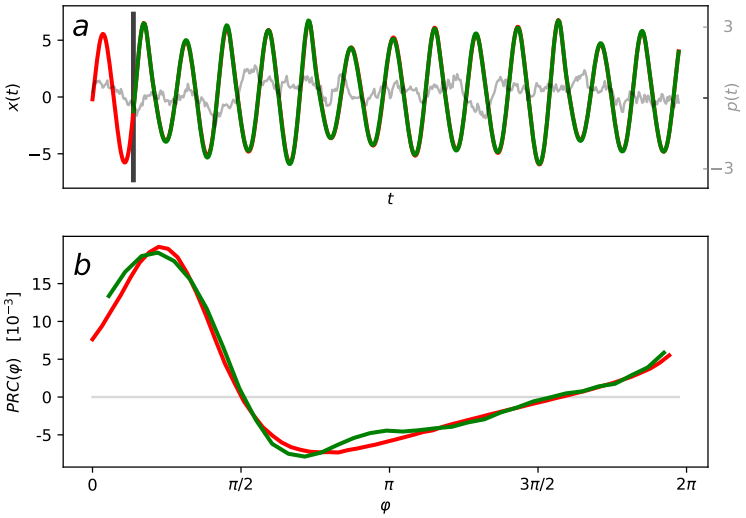


FIGURE 4.1: a: The training signal generated with Eqs. (4.2) in red and the RNN reproduced signal in green. Both have the same p-input, depicted with gray (scaled for being visually comparable to the signal). The vertical black line marks the beginning of the forecast. b: The true phase response curve of system (4.2) in red and the one inferred from the RNN in green.

the offset of the p-input $p(t)$. For this study we use a longer timeseries corresponding to 10000 natural periods, which is presented to the network during 1000 training epochs.

As a result, we find that the network reproduces the signal of the system perturbed by Eq. (4.4) well. Furthermore, we can estimate the bifurcation diagram from the network by feeding it different values of constant p-input, effectively setting the parameter b of the model (4.2) and observing the stationary signals, see Fig. 4.2(a). In the value range of the p-input (4.4) the diagram obtained from the RNN matches the true one closely. It reproduces simple oscillatory regimes, chaotic regimes and the period doubling bifurcation. Throughout the range of b the natural frequency (average frequency in the case of chaos) matches the true one closely, with mean deviation of 5×10^{-2}.

In chaotic regimes the maximum Lyapunov exponent [111] is an important measure as it quantifies the divergence of nearby trajectories in time. We estimate it from the RNN and plot it against the true values, see Fig. 4.2(b). This is accomplished by long time observation of the evolution of two nearby states, while re-scaling their difference to prevent them from diverging far from each other, see section 4.3.3 for further details. This can be an effective method for inferring the Lyapunov exponent from data, cf. [122, 123].

Now we go even one step further and test the prediction of the RNN when presented with an input outside the range of trained values. We train two networks on slightly modified p-inputs:

$$p(t) = \frac{1}{2}\exp(\pm|q(t)|) \tag{4.5}$$

This effectively splits the probability distribution of the p-input (4.4) in two at the value $b = 0.5$. One network is trained only on values smaller than 0.5 and the other only on larger ones. Then we perform the same prediction analysis as in the previous test, estimating the bifurcation diagram and the Lyapunov exponents accross the full range of $b \in (0, 2)$. It stands to reason that the predictions in regimes far from those presented during training will have little to do with the original system, but nevertheless it is surprising just how much can be deciphered from them. For example, in Fig. 4.3(a) a period doubling bifurcation occurs outside of the trained regime, as it does in the original system (although the critical values are shifted), and throughout the entire test range the system remains oscillatory (it does not settle to a fixed point). Not all features are reflected however, for instance in Fig. 4.3(c) in the chaotic regime outside of the p-input range the period-3 window is not observed.

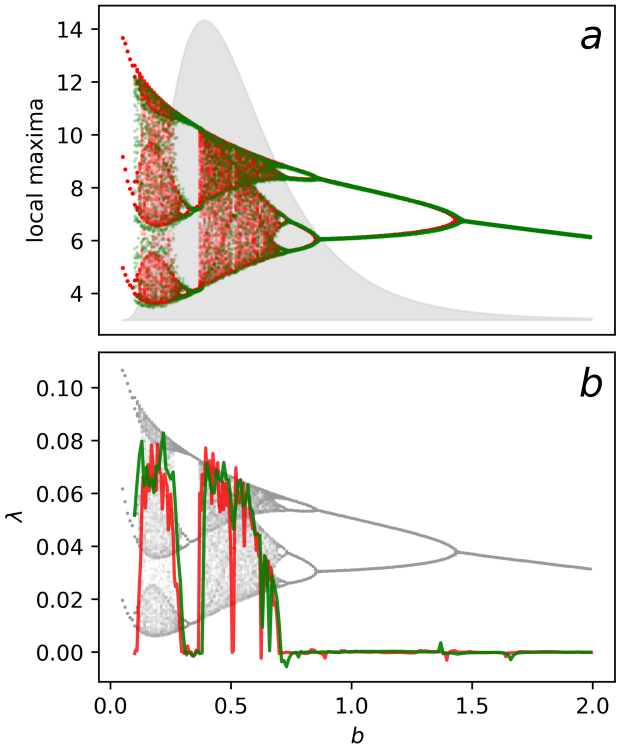


FIGURE 4.2: a: The bifurcation diagram of system (4.2) in red and the one inferred from the RNN in green. The probability density of the p-input is underlaid in gray (scaled for being visually comparable to the diagram). b: The Lyapunov exponents of system (4.2) in red and the ones inferred from the RNN in green. The true bifurcation diagram is underlaid in gray (scaled for being visually comparable). Note that the bifurcation diagram as well as the Lyapunov exponent range were reproduced with a single RNN trained on correlated noise p-input, Eq. (4.4).

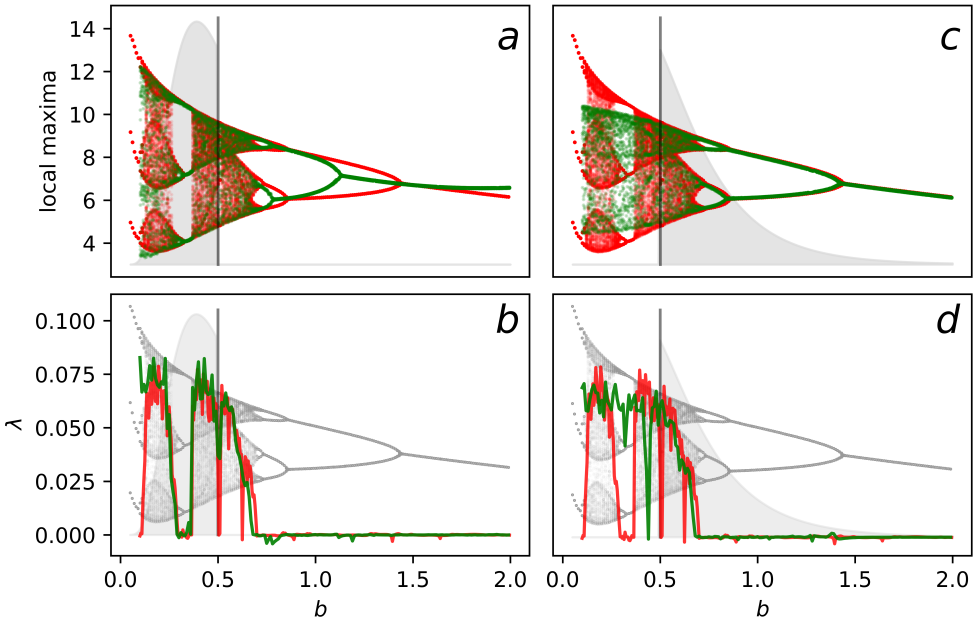


FIGURE 4.3: The bifurcation diagram and maximal Lyapunov exponent for system (4.2) perturbed by Eq. (4.5). On the left (a,b) the negative exponent (-) is used, therefore limiting the p-input values below 0.5, and on the right (c,d) the positive exponent (+) is used, limiting the p-input values to above 0.5, see the underlaid probability distributions in gray. Therefore, each side has a range of parameter values b that the RNN has not been presented with during training. In all subplots the true values of the system (4.2) are depicted with red and the RNN inferred ones with green.

FitzHugh-Nagumo oscillator - example of an excitable system

In the following two examples we want to study the power of RNN for two systems with different origin and dynamical behavior of oscillations. As a first important class we investigate an excitable system, namely the FitzHugh-Nagumo oscillator [6]:

$$\begin{aligned} \dot{x} &= x - x^3/3 - y + I_0 + p(t) \\ \dot{y} &= \sigma(x + a - by) \end{aligned} \tag{4.6}$$

where parameters are $\sigma = 0.1$, $a = 0.7$, $b = 0.8$ and $I_0 = 0.25$. For the p-input we use, as once before $p(t) = q(t)$, described by Eq. (4.3), with $\epsilon = 0.05$ and $\tau = 25.0$. The RNN is trained on timeseries comprising of 1000 spikes, over 500 training epochs. The time resolution is $\Delta t = 1$. For the excitable oscillations we compare how well does the model reproduce a spike train when presented with a novel p-input realization, see Fig. 4.4(a). Furthermore, we estimate the spiking frequency with respect to the input current I_0 (the p-input) and compare it to the true one, see Fig. 4.4(b). For low input currents I_0 the system (4.6) is quiescent, i.e. it does not fire and remains close to its fixed point. When I_0 is increased, a bifurcation occurs, a limit cycle is born and the system begins to spike regularly. The corresponding first-order phase transition is clearly inferred from the RNN, with the critical value of the input accurately predicted up to the order 10^{-3}. In addition, the estimated frequency values match the true ones closely, with mean deviation 10^{-2}.

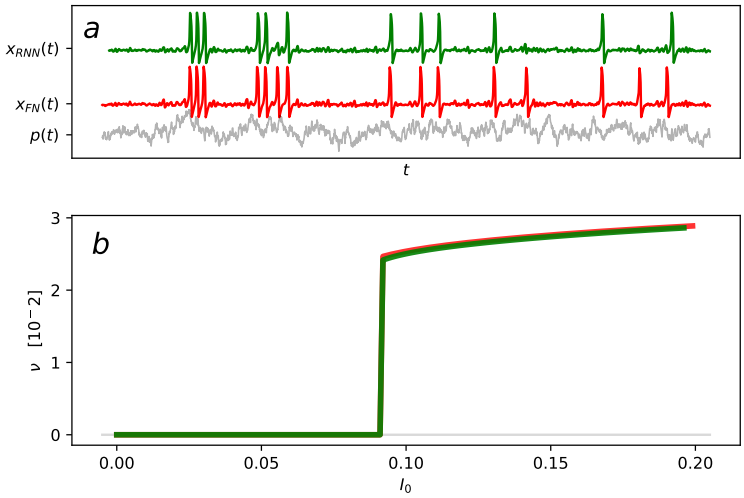


FIGURE 4.4: a: The training signal generated with Eqs. (4.6) in red and the RNN reproduced signal in green (shifted up for distinction). Both have the same p-input realization, depicted with gray (scaled and shifted for being visually comparable to the signal). b: The true spiking rate of system (4.6) in red and the one inferred from the RNN in green.

Mackey-Glass equation - example of a delay system

For our final study we briefly report on the RNN results for the Mackey-Glass equation [2], as a representative of time-delay systems:

$$\dot{x} = a\frac{x_\theta}{1+x_\theta^n} - bx + p(t) \tag{4.7}$$

where x_θ represents the time delayed variable $x(t-\theta)$, $a = 2$, $b = 1$, $n = 8$ and the time-delay $\theta = 2$. In this parameter regime the equation yields a stable limit cycle with a period-2 orbit, see Fig. 4.5. We use p-input $p(t) = q(t)$, Eq. (4.3), with $\epsilon = 0.005$ and $\tau = 1.0$. The length of time series corresponds to 5000 natural periods, over 500 epochs in batches of 100 time points with resolution $\Delta t = 0.15$. The dynamics is well reproduced with a mean deviation of 5×10^{-2}. Intuitively, a RNN seems to be suited well for modeling delay equations, since it has inherent delay. We conclude that for this important model class RNNs work well.

4.2.3 Amount of data and noise study

Any good statistics-based study includes a section on the dependence of the result on the amount of data provided and the sensitivity to noise - we do so in the following paragraphs. We present only results for the Roessler oscillator, Eq. (4.2). We train independent RNN models with different lengths of timeseries. We vary the amount of data supplied to the RNN in the following way: we keep the product of the timeseries length and number of epochs constant, thereby always introducing the same number of data points to the network (500000), i.e. we change the number of occurrences of the same points. The sampling rate is kept constant. We test the range from 15 to

1000 periods and measure the error of the PRC and signal, see Fig. 4.6. For each set of parameters 100 models are trained and evaluated. The PRC error is evaluated as the L_2 norm of the difference between the true and the reconstructed curve:

$$\int_0^{2\pi} \Big(PRC_{_{RNN}}(\varphi) - PRC_{_{TRUE}}(\varphi) \Big)^2 d\varphi$$

and similarly for the signal error:

$$\int_0^{\Delta} \Big(x_{_{RNN}}(t) - x_{_{TRUE}}(t) \Big)^2 dt$$

where we further have to determine over what interval we evaluate it, Δ.

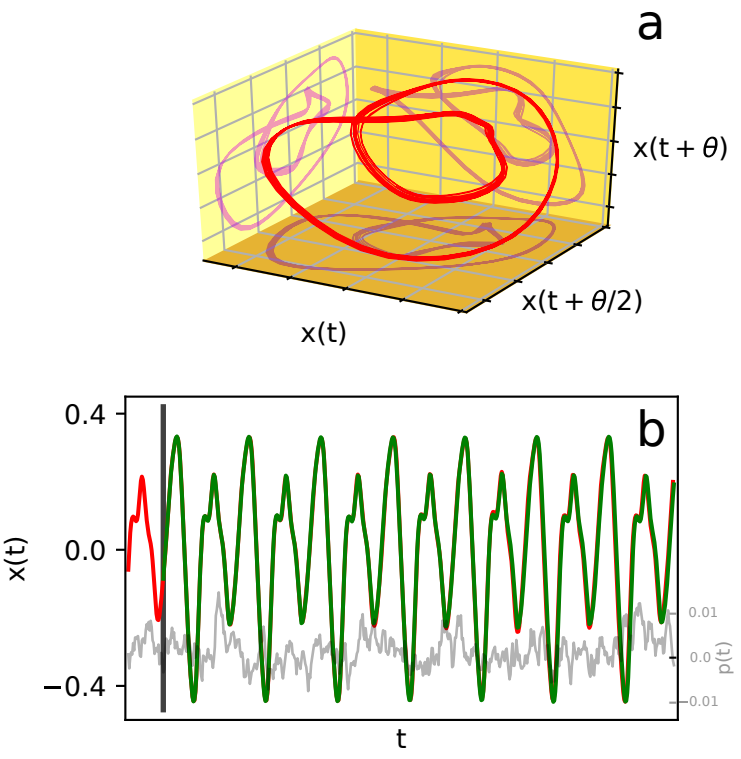


FIGURE 4.5: a: Delay embedded trajectory of system (4.7) in red, its two dimensional projections in purple. For the chosen set of parameters the system has a stable limit cycle, the variation is due to the p-input. b: The signal generated with Eqs. (4.2) in red and the RNN reproduced signal in green. Both have the same p-input realization, depicted with gray (scaled and shifted for being visually comparable to the signal). The vertical black line marks the beginning of the forecast.

The results of this evaluation are shown in Fig. 4.6. Here we also demonstrate the difference between LSTM and GRU cell types, underlining our previous remark on the poor results for GRU. For the LSTM cell, the PRC error is on average rather good and it clearly shows a dependence on the amount of data provided, approximately at 100 periods worth of data the error saturates around the value 0.1, see Fig. 4.6(a). In the case of GRU the PRC error is large and does not seem to improve with greater amounts of data, Fig. 4.6(b). That is not to say that GRU intrinsically cannot perform this task, it might just require a larger network to achieve the same effect - recall that we use the same number of nodes throughout this work. GRU was designed as a clever, faster simplification of the LSTM cell. It merges the hidden cell state into the regular cell state as well as merging several logical operations into fewer ones [118]. These simplifications are reasoned by its developers [117] but apparently noticeably impair the cell in performing our particular task.

The error of the signal undoubtedly should depend on the interval Δ over which it is evaluated. Even with a near perfect model the small errors build up and after a long time the true and reconstructed signals become incoherent, which means that with increasing Δ the error should grow. We see that for both cells in Fig. 4.6(c,d), although on average the GRU signal errors are significantly larger. As with the PRC error, the signal error decreases with the amount of data.

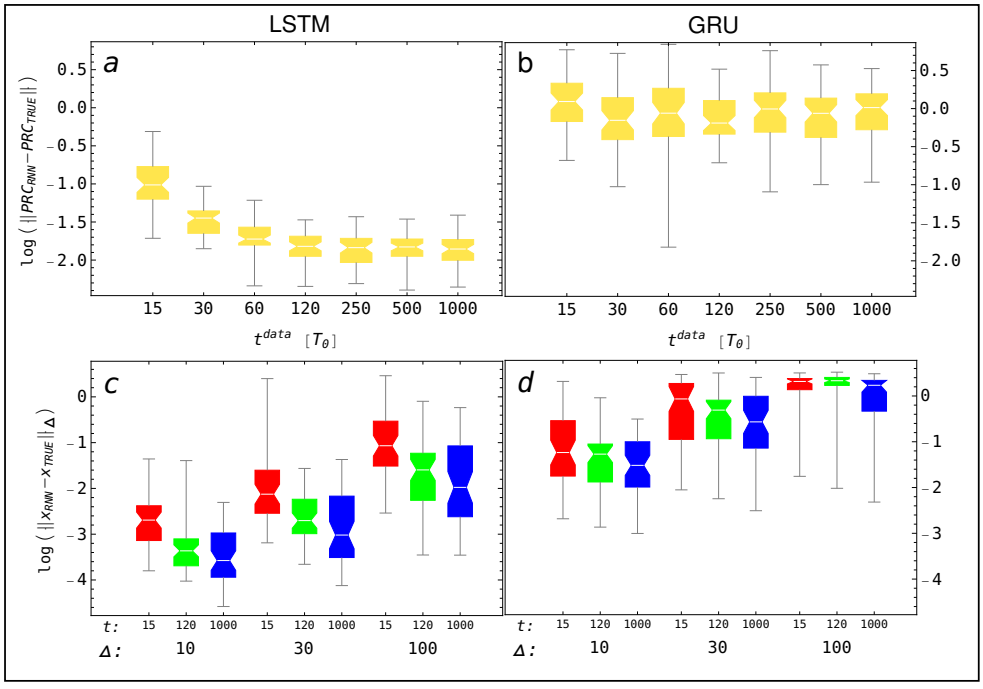


FIGURE 4.6: Comparison of data requirement for two different cells, LSTM left (a,c) and GRU right (b,d). In the top plots (a,b), the error of the inferred PRC with respect to the length of data provided, t^{data} (in units of the natural period T_0). In the bottom plots (c,d), the error of the reproduced signal with respect to the length of data provided t^{data} (15, 120 and 1000) for three different forecast lengths, Δ (10, 30 and 100).

Now for the robustness of the inference against measurement noise. We only present a basic study where we consider the Roessler system, Eq. (4.2) with p-input $p(t) = q(t)$, Eq. (4.3). We fix $b = 0.6$ which corresponds to a chaotic regime. Then, we add to each time point a random uncorrelated Gaussian number with mean 0 and standard deviation 1 to represent strong measurement noise, and train the network on the noisy signals. We introduce 10000 average periods worth of training data over 500 epochs. The network effectively extracts the relevant dynamics and reproduces the attractor well, see Fig. 4.7.

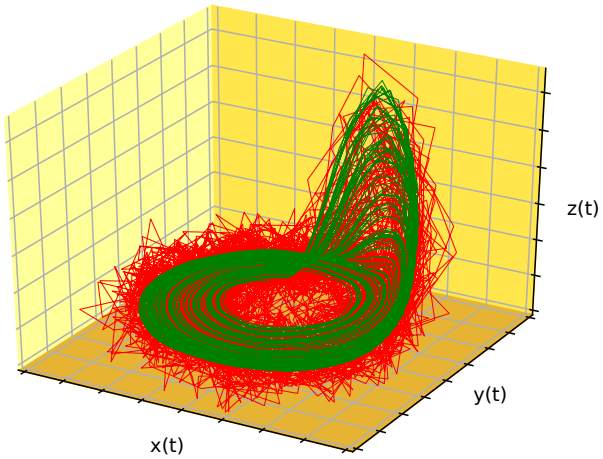


FIGURE 4.7: The training data in red and the RNN reproduced attractor in green.

4.3 Methods

In this section we specify the methods we used to evaluate the properties of oscillatory systems. For each property we write how we computed it from the equations as well as how we computed it from the RNN.

4.3.1 Natural period estimation

The period is measured as the time between two successive signal-threshold crossings from bellow when the system is unperturbed. From equations, the time of crossing is accurately estimated using the Hénon trick [11]. When estimating from a network, a linear interpolation from a point before and after the threshold crossing is used.

4.3.2 Phase response curve estimation

Firstly the natural period T_0 has to be accurately estimated, see section 4.3.1. Then the system in question is weakly and instantaneously perturbed at particular phases φ^*, i.e. at times $t^* = \frac{\varphi^*}{2\pi} T_0$ after the beginning of a period, $p(t) = \epsilon\delta(t - t^*)$. Then the evoked phase shift is evaluated as

$$Z(\varphi^*) = 2\pi \frac{nT_0 - \sum_{i=1}^{n} T_i}{\epsilon T_0} \tag{4.8}$$

where T_1 is the period in which the perturbation arrives and $T_2,\ T_3,\ T_4, ...$ the periods that follow. n counts how many periods we wait to evaluate the shift and since we are looking for the asymptotic shift n should be big enough that the PRC does not depend on it, in this chapter we used $n = 5$.

In the case of the network, the time for inputting perturbations is discrete and the best we can do is input perturbation $\epsilon/\Delta t$ where Δt is the time increment between two consecutive points in the unrolled RNN.

4.3.3 Maximal Lyapunov exponent estimation

For computing the exponents from the true system we use the standard technique, since we have the dynamical equations.

To estimate the exponent from the RNN a different approach is needed. Suppose we have access to all the variables of the original system $\vec{x} = (x_1,\ x_1, ...,\ x_{n_x})$. In such case the intuitive method can be used:

1. simulate a trajectory $\vec{x}$ for a long time so it settles to the attractor,
2. start a new trajectory $\vec{x^\dagger} = \vec{x} + \vec{p}$ with a small arbitrary perturbation $\|\vec{p}\| = \delta x$ and evolve both for a short time δt,
3. evaluate the deviation $\Delta = \|\vec{x^\dagger} - \vec{x}\|$,
4. renormalize the second trajectory for the deviation to have the same amplitude as the one we started with $\vec{x^\dagger} = \vec{x} + \delta x * (\vec{x^\dagger} - \vec{x})/\|\vec{x^\dagger} - \vec{x}\|$, but keep the direction of the perturbation the same so that the maximal exponents takes over in the course of several repetitions,
5. loop to step 3 and average the quantity $\frac{1}{\delta t} \log(\Delta/\delta x)$ which tends towards the maximal Lyapunov exponent.

Here $\|\cdot\|$ stands for the L_2 norm: $\|\vec{v}\| = \left(\sum_i v_i^2\right)^{1/2}$.

The more general approach concerns cases where we do not have access to all the variables but only a few, in the extreme case only one x_1 - common when dealing with real data. In such case the state of the system has to be characterized with several historical values, $\vec{w} = (x_1(t),\ x_1(t - \Delta t),\ x_1(t - 2\Delta t), ...)$, and then the algorithm above can be used as before. This is the case in section 4.2.2.

4.4 Discussion

The aim of this study was to test the predictive capacity of recurrent neural network applied to different oscillatory systems. One problem common to all oscillators is that the state space collapses to a low-dimensional manifold and therefore any reconstruction only allows the prediction on that inertial manifold. However, if perturbed we can achieve a much better understanding of the system around its attractor. We even can follow and predict a bifurcation outside the range of values which were provided by the data. This is a notable fact and it may as well work for other methods, like symbolic regression.

We have applied the method to a range of oscillatory systems, from a time-delay oscillator with a period-2 orbit (Sec. 4.2.2, Fig. 4.5), to an excitable system (Sec. 4.2.2, Fig. 4.4), and finally a chaotic attractor (Sec. 4.2.2, Fig. 4.2). We demonstrate that the trained neural networks can be probed for dynamical responses. As typical characteristics of oscillatory systems we estimated the phase response curve [55] (PRC), the spiking rate, and the maximal Lyapunov exponent [111]. Other quantities, such as the Floquet exponent [124], the amplitude response, the isochronal structure [7], synchronization properties [15] etc. could be estimated in a similar way. We can say that RNNs provide an effective way of estimating oscillatory properties from timeseries, cf. [60, 58, 50, 122, 123]. Our way of applying them to data is novel and should be explored further not only in the context of oscillations. It is, for example, not clear how well RNNs perform for scaling systems like turbulence.

Since the success of each machine learning method depends on data, we performed a statistical analysis on how the size of the training data set influences the inference. The training data required for an effective inference proved to be reasonably small, with only a few 10 periods sufficing for reliably estimating the mentioned dynamical systems quantities. We used two popular recurrent network cells in our study: the long short-term memory cell [116] and the gated recurrent unit [117]. The latter proved to be inferior in performing these tasks (at least for the same network size). We also tested the inference with the addition of measurement noise and it proved to be robust, see section 4.2.3.

Along with this publication, we (RC) published a Python software package, *OscillatorSnap* [125], available on the Python Package Index (PyPI) as: oscillator_snap. It contains most of the examples shown here as well as an array of high level functions for analyzing oscillatory systems, such as, a function that computes the phase response curve or the maximal Lyapunov exponent from dynamical equations as well as from a trained RNN model.

Acknowledgments

We thank Nicolas Deschle, Bastian Pietras, Thomas Kreuz, as well as the employees of Ambrosys, Markus, Franz, Tino, Maxim, Thomas and Greta for useful discussions. This work was funded by the European Unions Horizon 2020 research and innovation program under the Marie Skodowska-Curie Grant Agreement No. 642563 (COSMOS).

Chapter 5

Applications on data

We analyzed human gait recordings under anterior/posterior perturbations [130]. The perturbations were realized by stochastically varying the treadmill speed, the timescale of which was comparable to the average person's gait cycle. The recordings were used to train a recurrent neural network, which learned to reproduce the system's dynamics. This required no assumptions. The trained model exhibited a stable limit cycle, contrasting the notion of previous studies which suggest that human gait is chaotic. We inferred its phase response curve by probing it at different phases of the cycle, thus quantifying human gait's dynamical response to anterior/posterior perturbations.

Status: *to be submitted*
Associated software: OscillatorSnap @ Software section at the end.

5.1 Introduction

Gait is a primary form of human locomotion. It enables us to move from A to B even when the distance is substantially large. Walking and running come with challenges as we have to put our body's center of mass in close proximity of the area of support to avoid falling. Given that we can often manage that at ease, one might be tempted to conclude that the corresponding control process is a simple one. This, however, is not the case. Since humans have bipedal gait our body acts like an inverted pendulum and in order to control it the vast number of our muscles, each with non-trivial response characteristics, need to be activated in a concerted manner to produce stable gait.

Without going deeper into the anatomy and physiology of the human body, a brief literature scan reveals numerous computational models that mimic human gait and more or less account for the biomechanical and physiological reality. Most of them aim for adopting behavioral/biological properties to humanoid robots and do, in fact, encounter all but simple control strategies. Of course, one can qualify the gait these robots produce by simply considering them unstable whenever they fall. When studying human gait, however, deliberately letting participants of an experiment fall is rightfully marked unethical. Hence, the question arises: how to quantify gait stability? This question is not new but as of yet there is no consensus on how to quantify this stability properly [135].

The dynamical system theory provides many means to quantify stability properties of a system. Local, structural, and global stability of an attractor can readily be formalized but when it comes to determining corresponding measures from experimental data only local stability measures appear feasible and sufficiently robust to estimate. In gait analysis, for instance the largest Lyapunov exponent, Floquet exponents and multipliers have been used as proxies for stability [135, 134, 133]. Conclusions about general gait stability should be drawn with great care however, since these quantities measure local divergence of (movement) trajectories rather than the risk of falling (determining this certainly requires additional information about structural and global stability). Nevertheless, using the perspective of dynamical systems is the appropriate approach. Determining the aforementioned quantities is laborious due to statistical evluation of local divergence, and, more importantly, the quantities provide information about stability of the entire attractor. This is unfortunate because gait stability is known to vary substantially within a gait cycle: perturbations may have entirely different consequence when applied in the legs swing phase than in the stance phase.

In search for alternative assessment we approach this challenge from a different angle and train an artificial recurrent neural network to describe experimentally observed movement trajectories. Once trained, this network can be readily perturbed which allows for determining any arsenal of stability measures at ease, be that measures of local stability or, if the resulting model turns out to be a low-dimensional, stable limit cycle, measures to quantify this type of dynamics like the phase response curve. By explicating this approach we sought to illustrate the potential benefits for such a machine learning method to recover known and identify new dynamical properties of human gait.

5.2 Data description

We included data collected by Moore and coworkers [129], who measured 3D whole-body kinematics and ground reaction forces during treadmill walking; cf. Fig. 5.1. The data are available online via http://dx.doi.org/10.5281/zenodo.13030, [130].

Experimental procedures included trials during which pseudo-randomized accelerations in the belts speed served as perturbations along the walkers anterior/posterior direction. The maximum belt acceleration did not exceed 15 m/s^2. An example of the resulting speed fluctuations around a mean of 1.6 m/s, is shown in Fig. 5.2. The characteristic time of the fluctuations was comparable to the average gait cycle duration.

Kinematic recordings included 47 markers placed on anatomical landmarks using a 10 Osprey camera motion capture system paired with the Cortex 3.1.1.1290 software (Motion Analysis, Santa Rosa, CA, USA) and ground reaction forces were derived from force plates in an instrumented treadmill (R-Mill treadmill with dual 6 degree of freedom force plates, independent belts for each foot, along with lateral translation and pitch rotation capabilities, Forcelink, Culemborg, Netherlands). All signals were sampled at a rate of 100 Hz. For a more detailed description of the experimental protocol and data acquisition refer to [129].

FIGURE 5.1: The setup of the experiment. The vertical direction is defined as y, anterior/posterior direction as z and the mediolateral one as x. Figure adopted with permission from Moore 2015 (CC-BY-4.0) [129].

5.3 Approach

5.3.1 Signal selection

Since the current study was meant as a proof-of-concept for the usability of our recurrent neural network approach to 'measure' gait stability, we selected data from a single participant (20 year old male, height 168.8 cm, 68 kg); see [130] for participant #9. As shown in Fig. 5.2, we only included the position profiles of the pelvis marker and determined the corresponding 3D-velocity to minimize a possible low-frequency

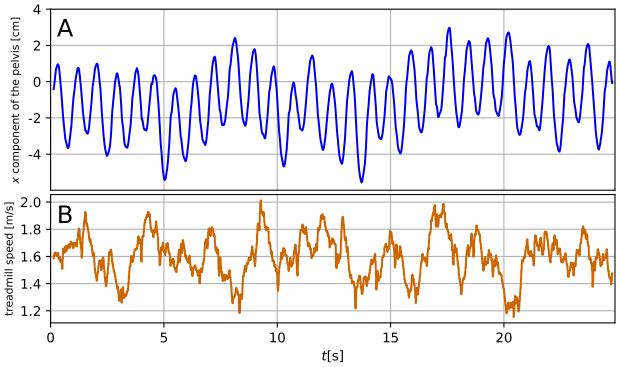


FIGURE 5.2: A: The vertical position of the pelvis in blue (average over all 4 pelvis markers) and in B: The treadmill speed in dark orange.

drift (whole-body displacement on the treadmill)[1]. We choose the pelvis marker because of its closeness to the bodys center-of-mass, i.e. its displacement/trajectory divergence can be considered informative about postural stability [131, 132].

5.3.2 Recurrent neural network

We employed a recurrent neural network with long short-term memory cells identical to the one described in Chapter 4. The network contained one hidden layer with 100 nodes and we employed sixty past steps. The inputs for each time step were the value of the treadmill speed $p(t)$ and the vector of the pelvis velocity $\vec{v}(t)$, while the output was the next vector of the pelvis velocity:

$$p(t), \vec{v}(t) \mapsto \vec{v}(t + \Delta t)$$

The time resolution $\frac{1}{\Delta t}$ agreed with the 100 Hz sampling rate of the data. The network was trained on the entire ten-minute walking session over 500 epochs with a learning rate of 0.03. The input data is depicted with blue in Fig. 5.3A.

5.4 Results

After training, the data left its imprint on the recurent neural network's dynamics. The trained network could then be investigated to reveal its dynamics, which should be reminiscent of the dynamics of the system from which we collected the training data, human gait. Because the network was trained on both dependent and independent variables (pelvis velocity and treadmill speed respectively) we could probe the network with different realizations of the treadmill speed. The signal generated with the true realization of the treadmill speed is depicted with green in Fig. 5.3B. Then, the network was presented with a constant treadmill speed, in effect to see whether it settles to an attractor and what type of attractor that would be. The network was shown to settle to a limit cycle. This was apparent from the spatially isolated

[1]Velocities were determined via a two-point derivative scheme. Although this procedure substantially amplifies measurement noise, we did not apply any smoothing since the training of the recurrent neural network can effectively reduce measurement noise; see Chapter 4, section 4.2.3.

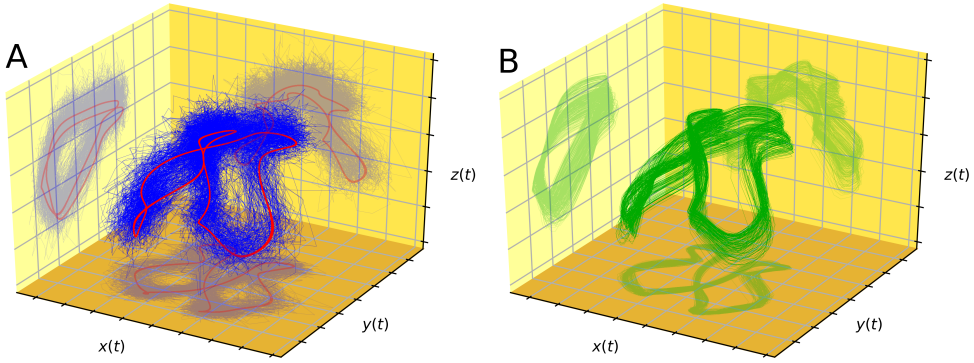


FIGURE 5.3: A: The velocity of the pelvis in blue (determined with a simple two point derivative scheme) and the limit cycle of the inferred neural network model in red. B: The trajectory of the neural network model perturbed with the true treadmill velocity variation.

and periodic trajectory, depicted with red in Fig. 5.3A. By probing the network for divergence of nearby trajectories we quantified the largest Lyapunov exponent to be 0, i.e. two nearby trajectories never diverge over time, further indicating that the network's attractor is a limit cycle.

5.4.1 Phase response curve

Since the trained model seemingly demonstrated a stable limit cycle, we determined its corresponding phase response curve. For this, we probed the trained model for asymptotic phase shifts when perturbed, as outlined in Chapter 4, Section 4.3.2. The obtained phase response curve can be seen in Fig. 5.4, where characteristic events of a gait cycle such as heel-strike and toe-off moments are indicated with vertical lines for reference. Further for reference in the figure we also included the 3D-position of the pelvis throughout the cycle with color lines, these positions were determined by integrating the limit cycle velocities. The frequency of the vertical and anterior/posterior direction was twice of that in the mediolateral direction resembling steps vis-à-vis stride frequencies. This agreed with the PRC's resemblance of the first half period $(0, \pi)$ to the second half $(\pi, 2\pi)$. In fact, the PRC displayed three phases most sensitive to asymptotic phase shifts: one corresponded roughly to the moment of toe-off, the other one to the maximum vertical position and the third one to the most anterior position. The phase response curve turned out predominantly positive implying that, as one could expect, the gait cycle duration was reduced whenever a perturbation was in the direction of walking and slowed down in the opposite direction.

5.5 Discussion & Conclusions

Our recurrent neural network model could be readily trained to the 3D-velocity profiles of the pelvis of a single participant walking on a treadmill with randomly perturbed speed. We found that the inferred model contained a stable limit cycle (Fig. 5.3) contrasting the notion of previous studies which suggest that human gait is chaotic [133, 134, 135, 136]. This does not prove that gait is governed by limit cycle

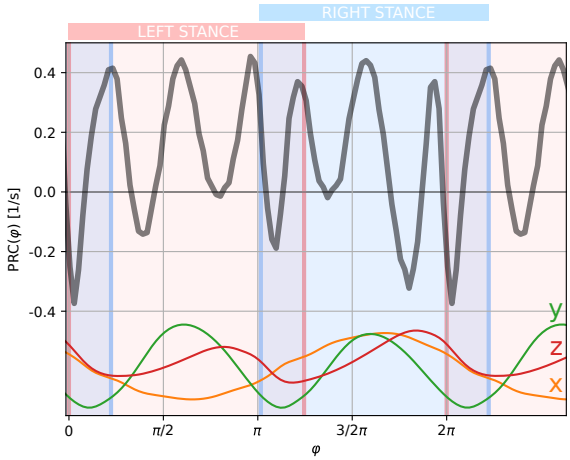


FIGURE 5.4: The phase response curve of the gait oscillator for anterior/posterior perturbations, depicted as a thick gray curve. The phase duration of the left stance is shaded with pink and of the right with blue, making the double support phase purple. The heel-strike and toe-off moments are marked with vertical lines, color coded for left and right. In the lower panel the positions (not velocities) of the pelvis at each phase are shown in colors, orange, green and red representing x, y, and z respectively.

dynamics but merely suggests that the dynamics that best describe perturbed gait within the framework of a recurrent neural network model are limit cycle oscillations. Further studies are needed to definitively answer such questions.

Given its limit cycle characteristics, the network model allowed for estimating the phase response curve with respect to anterior/posterior perturbations (Fig. 5.4). The phase response curve is a universal property of an oscillator. It is a good candidate for quantifying gait characteristics such as stability, since it by definition quantifies the sensitivity to perturbation, albeit this correspondence is yet to be tested further. Alongside the phase response curve, our approach allows us to measure other phase dependent quantities as well, such as phase dependent scaling exponents. We recommend the further use of inferring dynamical properties via probing a generic model trained on data, particularly in cases where perturbing factors can be measured. These perturbations can be incorporated into the model which allows for isolation of the system and the dynamics of interest.

While we have quantified the response of human gait to anterior/posterior perturbation this certainly is not the only factor influencing it. The gait pattern can vary in many ways, depending on constraints imposed by stability, energy cost and attention, as well as inherent variability due to motor noise. Most of these factors can not and were not controlled in the data we analyzed. This is also why the reconstructed trajectory does not match the original data completely (Fig. 5.3).

In conclusion, neural network model inference proved powerful for estimating dynamical properties that are generally difficult to statistically distill from data. Our preliminary results on the analysis on human gait recordings indicated the presence of a stable limit cycle in the model, which is surprising and should be investigated further with larger datasets and additional methods in order to better understand the fundamental dynamical driving of human gait.

Chapter 6

Discussion

This thesis summarises several methods for inferring dynamical properties of oscillatory systems from observations of their time series. In Chapter 2, I introduced a method aimed at estimating the phase response curve from observations of a single self-sustained oscillator [50]. The method was designed for spiky oscillators, and therefore requires times of spikes (or other events) as well as the knowledge of the perturbing force. It yields an estimation of the phase response curve and natural period as well as of the asymptotic phase as a function of time. In Chapter 3, a similar idea was implemented in a method designed for inferring the connectivity of a network of pulse-coupled units [51]. It requires the spike times (or other events) of all units and yields the weighted connectivity between units as well as the properties of units themselves, namely, their natural period and phase response curve. Chapter 4 is slightly different but devoted to the same theme. It introduces a scheme with artificial neural networks that allows the inference of many dynamical properties [52], and is applicable to many different kinds of oscillatory systems, from self-sustained, to excitable and even chaotic. In Chapter 5, I applied the scheme of Chapter 4 to human gait data. An artificial neural network model was trained on gait data and contrasting the common belief, it exhibited non-chaotic dynamics. From the model, the phase response curve was extracted. In this chapter I discuss extensions and applicability of my methods as well as their expected behaviour in settings that break their assumptions.

6.1 Improved phase approximation

In the introductory Section 1.2.2 I outlined how to approximate the phase from signal observations. Here I discuss how one may improve its accuracy as well as quantify the magnitude of the error. I refer to the approximation introduced in the Introduction 1.8 as the zero approximation $\varphi^{(0)}$, since it disregards the isochronal structure. For a more accurate approximation the isochronal structure has to be considered, which implies considering the amplitude. Amplitude can be defined in different ways. Note that there is a trade-off between amplitudes that are "easy" to determine from time series, and amplitudes that yield simple dynamical equations [126, 17, 138]. An example of the former is:

$$A(R,\theta) = \frac{R - R_0(\theta)}{R_0(\theta)} \tag{6.1}$$

where $R_0(\theta)$ is given by the corresponding radial coordinate of the limit cycle. This amplitude is defined to be 0 on the limit cycle, positive outside and negative inside.

An example for the second case is amplitude that for autonomous dynamics follows the equation:

$$\dot{A} = \mu A \tag{6.2}$$

Regardless of which exact definition of amplitude is considered (definition for simple evaluation (6.1) or neat dynamical equations (6.2)), its incorporation can improve the accuracy of the estimated phase. I refer to the approximation where amplitude appears linearly as the linear approximation $\varphi^{(1)}$:

$$\varphi^{(1)}(R,\theta) = \varphi^{(0)}(\theta) + \alpha(\theta)A(R,\theta) \tag{6.3}$$

where $\varphi^{(0)}$ is the zero approximation of the phase (1.8) and α is a function of the protophase θ which holds the information of the inclination of the isochrons on the limit cycle. How to estimate this function from a general signal is an open problem.

To improve the approximation further away from the limit cycle, one possible expansion is:

$$\varphi(R,\theta) = \varphi^{(0)}(\theta) + \alpha A + \beta A^2 + \gamma A^3 + ...$$

where all α, β, γ and further terms are functions of the protophase θ. However, one faces the same problem of not knowing how to estimate them in general.

6.1.1 Error of the phase estimated from time series

From the difference between the zeroth (1.8) and the linear (6.3) approximation one can realize that the zeroth approximation (even without considering any errors from the embedding) has an error of the order $\mathcal{O}(A)$, i.e. first-order in amplitude. This does not seem too bad, particularly since amplitude is asymptotically stable and therefore amplitude variations are typically not very large. However, when evaluating the phase derivative in such a way one makes an error of the order $\mathcal{O}(\dot{A})$, i.e. first order in the amplitude's derivative. Note that, even for vanishing perturbations the amplitude derivative can in general be of the same order of magnitude as the phase derivative, calling for caution when phase derivative is evaluated without any *a priori* knowledge of isochrons.

6.2 Why the use of integrated phase form?

In Chapters 2 and 3, I assumed that the system can be described with the Winfree phase equation:

$$\dot{\varphi} = \omega + Z(\varphi)p(t) \tag{6.4}$$

The idea for using the integral form equation:

$$2\pi = \omega T + \int_T Z(\varphi)p(t)dt \tag{6.5}$$

served to develop a method for evaluating the phase response curve using nothing but the timings of the units' spikes, which are all assumed to occur at a particular phase. For continuous time series, one may try evaluating the phase via signal embedding (Section 1.3) and a protophase-to-phase transformation (Section 1.2.2) and numerically evaluate the phase derivative, which would suffice to infer the PRC directly from equation (6.4). However, this intuitive procedure may not work in general.

The instantaneous phase derivative evaluated in this way has an error of the same order of magnitude as the phase derivative itself, see Section 6.1.1. This is due to the isochronal structure of the oscillator. The magnitude of this error is particularly easy to miss since the actual values of the phase (as opposed to its derivative) can be evaluated with a relatively small error even when ignoring the isochronal structure.

6.3 The effect of unknown inputs on the quality of inference

In this section I discuss the behaviour of the method of Chapter 2 when the system in question contains perturbations that are not measured. I consider a two dimensional limit cycle oscillator:

$$\begin{aligned} \dot{x} &= \quad y - \sin(y)x/4 + u(t) \\ \dot{y} &= -x + \cos(x)y/4 + p(t) \end{aligned} \tag{6.6}$$

where $p(t)$ is the known input and $u(t)$ the unknown one. I set the unknown input to be an Ornstein-Uhlenbeck process [26] (4.3) with fixed correlation time $\tau_u = 0.02$ and amplitude $\epsilon_u = 0.2$. Then, I introduce different strengths of the known input ϵ_p and record the error measures Δ_ψ, Δ_{ψ_T} (2.12, 2.13). These error measures, introduced in Chapter 2, quantify the irregularity of the time series and how much of it is explained by the inferred model. Since the method takes discrete time events as inputs, they are generated by signal threshold-crossings. I use long time series of roughly 1500 periods for the inference. The irregularity of inter-event intervals Δ_{ψ_T}, grows with the strength of known forcing ϵ_p as expected, and has a non-zero value at $\epsilon_p = 0$ due to the unknown forcing ϵ_u. The error of inference Δ_ψ, that corresponds to the deviation of the recorded inter-event intervals from our considered model, stays close to the value of irregularity due to unknown forcing only (Δ_{ψ_T} at $\epsilon_p = 0$). This makes sense since we can only infer relations for which we have observations. For higher values of the known forcing ϵ_p, the error Δ_ψ begins to grow. This is likely due to the diminishing validity of the linear phase model (6.4) at higher amplitudes, see Figure 6.1. What is perhaps more surprising is that a similar picture is true for a chaotic oscillator without any unknown inputs. I have mentioned before that while chaotic oscillators strictly speaking cannot be described purely with phase dynamics, it might be reasonable to approximately describe them with something representing the phase and an intrinsic stochasticity accounting for diffusion [137, 127]. This test provides an argument for the validity of this description, see Figure 6.1. The oscillator in the test is a Roessler oscillator [4]:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay + p(t) \\ \dot{z} &= b + z(x - c) \end{aligned} \tag{6.7}$$

$a = 0.03$, $b = 1.5$, $c = 100$ and $\tau_p = 0.2$, and the input $p(t)$ is an Ornstein-Uhlenbeck noise as before.

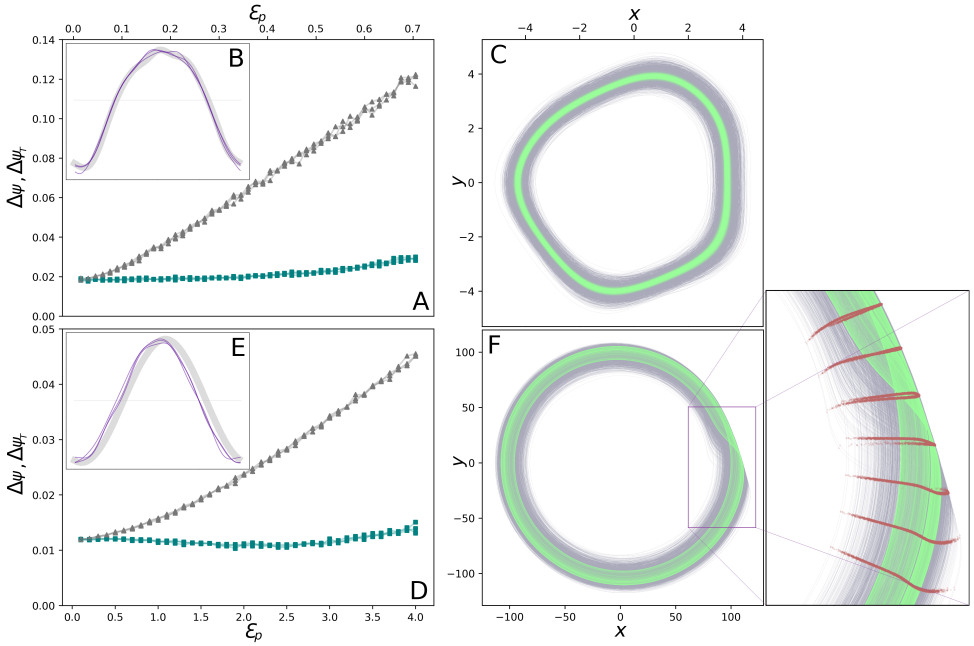


FIGURE 6.1: Comparison of a chaotic oscillator with a noisy limit cycle oscillator, using the method of Chapter 2. A,B,C: a limit cycle oscillator with unknown input and in D,E,F: a chaotic oscillator. In A and D: the inference errors with respect to the known input strength ϵ_p, Δ_ψ with blue squares and Δ_{ψ_T} with gray triangles. For each ϵ_p three measurements were made. In B and E: the true phase response curve in thick gray (for the chaotic oscillator I took the curve of a limit cycle with similar parameters) and three reconstructed ones in purple, corresponding to values $\epsilon_p = 0.15, 0.3, 0.45$ for A and $\epsilon_p = 1, 2, 3$ for D. In C and F: the trajectory of the system without known input in green and the trajectory with known input in gray. In C: $\epsilon_p = 0.3$ and in F: $\epsilon_p = 2.5$. Panel F also has a zoom into the twisting part of the attractor, there I depict the points corresponding to the same value of the phase in red. It is possible to see the folding of the Roessler system.

6.4 Applicability of my methods to systems that deviate from phase dynamics

The methods from Chapters 2 and 3 assume phase dynamics in the Winfree form (6.4). This equation is the first-order phase expansion for a limit cycle oscillator perturbed by a scalar force $p(t)$. As such, it is valid only for weakly perturbed limit cycle oscillators that stay within the vicinity of their limit cycle. For stronger perturbations or dynamics different from limit cycle oscillations the validity of the equation diminishes which is reflected in the poorer quality of reconstruction using my methods. Here I discuss my methods' applicability to systems that deviate from the Winfree phase description. The discussed cases will be:

- a limit cycle oscillator with a strong perturbation,
- an excitable oscillator,
- a chaotic oscillator.

I discuss each case individually.

6.4.1 Strong forcing

Forcing strength is vaguely defined. What I consider weak, is forcing that keeps the variation of successive rotations low and allows the system to be described with a phase equation, where the phase variable grows monotonically. Forcing that breaks any of the mentioned conditions will be considered strong. There is a wide range of strong forcings, and certainly any oscillator can be forced so strongly that my methods will fail. However, at what point they begin to fail depends on the specifics of the oscillator. For example, one oscillator I have tested my methods on is the Morris-Lecar neuronal model [65]. I was able to infer its dynamics even from what is considered very strong forcing, where the variation in the periods exceeds an order of magnitude and the asymptotic phase at times clearly decreases for an extended period of time, see Fig. S2.2 for an example. This oscillatory model is so robust due to its dynamics. It has a limit cycle on which the flow slows down significantly in a narrow region, see its isochronal structure in Fig. 1.4C. The oscillator spends the majority of its time in this narrow region and only rapidly whirls by the rest of the cycle. When the oscillator is moving fast, it is less prone to perturbations. Therefore most of the forcing happens in that narrow slow region, where the forcing has a simplified effect.

In general, a strongly forced oscillator cannot be well described by a linear phase equation. However, a self-sustained oscillator might still have a clearly defined asymptotic phase, even far from the limit cycle, that is, as long as the perturbations do not lead the system out of the limit cycle's basin of attraction. As such, the system may not be well described by the linear phase equation, but might still be well described with a phase equation if a few more terms and/or variables are considered. The natural extension is to include amplitude [126, 17, 138].

6.4.2 Excitable systems

For an excitable system there is no way to define the asymptotic phase that would be analogous to the phase for limit cycle oscillations. The autonomous system does not

have a natural frequency. However, it has a characteristic time, namely the excursion time t_e, which is the time it takes for the autonomous system to go from the excited state back to the resting state. As said, the excursion time cannot be non-arbitrarily defined (Section 1.1.2), yet together with a definition it still describes the system dynamics sufficiently well. In a way, an excitable oscillator acts like a hybrid system: from the start of the excitation until it reaches the resting state it can be treated rather like a self-sustained oscillator (with the excursion trajectory being like a part of the limit cycle) and the time it stays in the resting state it acts, for practical intents and purposes, like a simple stable fixed point. Thus, one can consider the two different modes of the oscillator separately. If the system spends very little time in the resting state, then my methods might work well. Admittedly, most excitable models never yield regimes where my methods could be applied successfully, even if they were in their most coherent oscillatory regime [5].

6.4.3 Chaotic oscillators

A rigorous concept of the asymptotic phase cannot be associated with a chaotic oscillator (Section 1.2). The positive Lyapunov exponent causes divergence in the system, which ambiguates the concept of phase. Therefore in general, the methods of Chapters 2 and 3 will not work well. However, for an oscillator in a regime that is strictly speaking chaotic but is not far from limit cycle oscillations (e.g. a system close to a bifurcation to chaos) the divergence might be weak and therefore some effective phase could still allow for a decent representation of the system with a phase equation. An interesting way to imagine this is by describing the chaotic oscillator (e.g. Roessler) with a phase variable and intrinsic noise that accounts for the phase diffusion [137, 127]. This scenario is equivalent to having a true phase oscillator forced by an unknown input, and as discussed previously, this situation breaks the inference if an unknown input is dominant, see Section 6.3 and Figure 6.1.

6.5 The maximal information content of data

In this section I discuss the limits of information content of data. Data can only contain information on the region of state space it occupies. This means that from arbitrarily long time series of a system on its attractor one cannot make any definite conclusions on the dynamics in the surrounding state space. Assumptions can aid in bridging some information gaps, but they should be used with caution, as spurious assumptions can lead to wrong conclusions.

As an example, let us consider finite dimensional ODE systems, $\dot{\vec{x}} = \vec{f}(\vec{x}), \vec{x} \in \mathbb{R}^N$. An arbitrarily long time series of the complete state $\vec{x}(t)$ contains complete information on the model that generated it. What about a partial state, for example only a scalar variable $x_i(t)$? The Taken's embedding theorem [25] ensures that in general this is the case. An exception to this are decomposable systems, called skew systems, where some variables are not influenced by the rest and therefore have no

causal relation with a part of the system. For an example of such a system, consider:

$$\begin{aligned}\dot{x} &= f_x(x, y)\\ \dot{y} &= f_y(x, y)\\ \dot{z} &= f_z(x, y, z, w)\\ \dot{w} &= f_w(x, y, z, w)\end{aligned} \tag{6.8}$$

and suppose it has a chaotic attractor. While the z and w are generic variables that contain all the information, x and y do not, since they form a disconnected sub-system, which is two dimensional and can therefore only be oscillatory and their model would certainly not have a positive Lyapunov exponent.

Now let us consider perturbed systems. Imagine a system with an attractor for which we have observations of the autonomous dynamics. So long as our observation starts in the basin of attraction, the system will relax to its attractor. Unless provoked, it will not leave it, which means that arbitrarily long observations of the autonomous dynamics will contain little to no information of the dynamics around the attractor. The situation changes if the system is perturbed. Then, depending on which parts of the state space the system visits, the observations of the perturbed system may contain information on the dynamics beyond the attractor. The perturbation however, introduces a new twist to the puzzle: the Takens' theorem does not apply to perturbed systems. This makes the field of inferring dynamical properties from partial observations of systems under perturbation still an open research area.

This leads me to pose some important questions of the field:

- Can dynamics always be inferred from observations of a perturbed limit cycle (the perturbation is also measured)? This is not an easy question and I cannot answer it in its generality, but it is likely true if the perturbation is variable enough. It can be hard to find even specific methods that are effective. The method I present in Chapter 2 is designed for the case of an oscillator that follows the linear phase equation (6.4). The method I present in Chapter 4 is more general because it leaves the problem of inference to a more versatile model - a neural network. Despite successfully testing on several examples, it is not clear whether this can always be achieved.

- Is it possible to infer the phase dynamics of a perturbed oscillator from observations of the protophase (as defined in the Introduction 1.2.2)? I can answer this, and in general the answer is no. This is due to the minimal error one makes when evaluating the asymptotic phase from protophase, see Section above 6.1.1. The exception to this are oscillators with radial isochronal structure, in their case the angle protophase is bijective to the phase, and that bijection can be reasonably well estimated.

6.6 Inferring connectivity from data that do not fulfill assumptions

In Chapter 3, I introduced a method for inferring connectivity of a network of pulse-coupled oscillators. It is based on the method presented in Chapter 2 so it has the same assumptions: the oscillators are to have stable limit cycles and the perturbations

are to be weak enough to allow a phase dynamics description. Further, I assumed that there is no delay in the coupling. When these assumptions are not fulfilled the inference will have some artifacts.

Let us discuss each case separately. First, **if the delay is non-negligible**: suppose each pulse emitted from unit j reaches unit i after some time τ_{ij}. If the coupling is rather weak, then this will not disrupt the inference of my method much. It will shift the phase response curve, but the strengths of the recovered connections will be roughly the same. It is when the coupling gets stronger that greater errors occur. In order to correct for that one has to incorporate time-delay into the model. This is feasible if one considers that all connections have the same delay. Then, one may do a one dimensional search over the different delay values and take the value that best fits the data, i.e. that yields the smallest errors of reconstruction (2.12). By the same principle one could also do it in the case where each connection has a different delay value, however, then the search would be higher dimensional rendering it less feasible.

Second, if the individual oscillatory nodes are either chaotic or excitable. **If they are chaotic**: several factors will affect the inference, most notable being the structure of the attractor and the value of the largest Lyapunov exponent. For a simple loop attractor, e.g. Roessler system, larger values of the Lyapunov exponent will disrupt the inference more. **If they are excitable**: one can treat the oscillators as being in roughly two regimes: the resting state, where the system remains bounded in a part of the phase state, and the excited state, where the system enters an oscillation. The time it takes the system to go from the excited state to the resting state is called the excursion time t_e and the time it spends in the resting state the resting time t_r. During the excursion time the system behaves very similarly to a self-sustained oscillator. Thus, if the resting times are on average much smaller than the excursion times, $t_r \ll t_e$, the inference might work reasonably well. It is in general difficult to find such conditions with excitable systems, even in regimes of coherence resonance [5] where the signal is most coherent, the resting times are usually considerable. I expect that the inference for excitable systems will be on average inferior compared to chaotic systems.

For a completely general system, it is difficult to expect a valid quantitative inference from my method that assumes particular dynamics. Nevertheless my method will likely detect connections between signals that are correlated and possibly even distinguish causal relationships.

6.7 Too few data

When data is scarce, errors appear. This is natural, some features require large amounts of data and shorter time series simply do not contain all the information required to determine them accurately (cf. Section 6.5). A paradigmatic example of this fact is the determination of the note pitch (frequency). Humans can gauge pitch with our hearing sense (between 20-20000Hz), e.g. musicians tuning their instrument need to be able to precisely sense the frequency of a note. To achieve this they have to listen to it for a "long" time (in this case seconds). This has foundations in physics, if we want to determine the frequency of a signal accurately we have to observe it for a long time. It is not just a matter of having a better measuring device, this is a fundamental limitation that the length of time series determines the accuracy of the

signal's frequency; that is why the Fourier transform of a short package of oscillations yields a wide distribution of frequencies. Similar principles apply to my methods - for a more accurate prediction longer time series are needed.

For the method of Chapter 2 less data translates to poorer accuracy of the phase response curve prediction. If some ranges of the phase are not stimulated enough during the limited time series, then those ranges will be poorly determined. This means that in those regions the minimization cost function is not affected much and therefore, one expects to see high arbitrary variations there. In such cases it can be beneficial to use fewer harmonics to describe the phase response curve. That way the arbitrary variation is limited, albeit at a price of a smaller solution space.

For the method of Chapter 3, the features most sensitive to limited amounts of data are the phase response curves of individual nodes. Connectivity inference seems to be much more robust and often accurate even if the phase response curves are not. Of the connectivity features, the most sensitive are identifications of non-existing links between nodes that are strongly correlated due to effects of common forcing and links that actually have strong causal interaction but not via a direct link (see Scheme 1.1 in the Introduction). My method is most likely to make false conclusions in these cases.

The method of Chapter 4 suffers the drawbacks of artificial neural networks, which is first and foremost overfitting in the case of limited data. The topology of the networks should be chosen appropriately to minimize the possibility of overfitting. This can be done by making the network more sparse for example (commonly referred to as dropout). It is conceptually a similar idea to using fewer Fourier harmonics for the method of Chapter 2. Overfitting in this case would mean that the network learns to reproduce the limited time series too closely and because of that fails to grasp any generic properties of the system, e.g. on a short chaotic time series it might just learn to reproduce a purely periodic signal. Inferring dynamical properties from an overfitted network like this results in arbitrary errors, e.g. if the network reproduces a purely periodic signal its maximal Lyapunov exponent will be 0.

6.8 Machine learning with neural networks

6.8.1 Hyperparameters of neural networks

The typical hyperparameters of neural networks are the number of nodes and the activation function. The choice of the activation function is in general not trivial but it was for my purposes less crucial. What was crucial was the size of the network. Smaller networks with fewer nodes cannot reproduce very complex dynamics. For example, take a simple feed forward neural network that has only one node in one of its hidden layers. Such a "bottleneck" causes the network to only be able to learn linear relations, regardless of the number of layers or their number of nodes. In general, it is difficult to specify which features a neural network gains with each additional node. Here, it suffices to say that the complexity of the network grows exponentially with additional nodes. Choosing the proper size of the network is difficult since the network has to be big enough to be able to reproduce the desired features, but not too big to overfit. Also, larger networks are computationally expensive. The hyperparameters are therefore typically determined via trial and error.

6.8.2 How to know the inference is correct when using machine learning?

Suppose we train a neural network to reproduce some time series and it reproduces them well. Then we perform an active experiment on the trained model to obtain some characteristics, like for example the phase response curve. How can we test whether that inference is proper? If the signal is well reproduced that should already be a good indicator that it is correct. But what could happen is that a different model reproduces the signal equally well but has different characteristics. In that case one can conclude that the signal does not contain enough information to determine the property one wants to infer (see Section 6.5 on data information content).

6.8.3 The complexity of the trained model

As shown in Chapter 4, Section 4.2.3, the recurrent neural networks when trained on a noisy signal filter out fast noise. They do this by only learning robust features. How can we tell which features are robust and which ones get filtered out? Certainly it depends on the size of the network. If one takes a network too small to produce chaotic dynamics, then if that model is trained on chaotic data it will filter out part of the complexity that gives the system its chaotic dynamics (similarly to how only two first-order ODEs cannot yield chaos). So the networks have to be big enough to structurally allow the complexity of the dynamics we are interested in. With the same idea in mind we can deduce how the networks filter out noisy features. It can be expected that as bigger and bigger networks are considered, less and less features will be filtered out, until even the noisiest features will leave their imprint on the network. Typically one does not want very noisy features to contribute, but that is generally not a problem since the network would have to be very large to see the reproduction of very complex processes that resemble stochastic noise.

In Chapter 5, Section 5.3.2, where I trained a recurrent neural network on recordings of human gait the trained model was not chaotic, it had a stable limit cycle. This was unexpected since many scientist who study gait consider it to be chaotic [133, 134, 135, 136]. The neural network filtered out any dynamics that can be considered chaotic. The possibility that the network might not be able to support chaotic dynamics can be ruled out, since I have used 100 nodes and before I was able to infer chaos with only 32 nodes, see Figure 5.3. In either way, this does not definitively prove that gait is indeed non-chaotic, it is merely an empirical argument.

6.8.4 Inferring connectivity with neural networks

In Chapter 4, I have used neural networks to infer the dynamics of a single oscillator. This was accomplished by training the network to reproduce the signal of a perturbed oscillator and then inferring dynamical properties with an active experiment on the trained model. The shape of the input and output of the network is completely adjustable. The training scheme is the same whether we train a network to reproduce a single signal, or several ones. A natural attempt for inferring the connectivity of a network of oscillatory units with neural networks would therefore involve training one neural network on reproducing signals from all the nodes. If the network learns to reproduce the signals well, a number of basic active experiments can be applied to infer the connectivity, such as, setting all but two units to give zero output and

evaluating the effect the two units in question have on one another. In practice, as the size of the network gets larger this direct approach might quickly become too demanding in both computation time and the amount of data required, however it is feasible for a few units.

6.9 Conclusions and outlook

Oscillatory systems are found in nature throughout all spatial and temporal scales, and while some present themselves for easy study, most of them come embedded in their environment, coupled to their surroundings and not easy to isolate. It is for these oscillatory systems that we require methods of inferring properties from observations of the system in its natural environment only. While some particular problems of this nature have been solved in their generality, such as topological attractor reconstruction for unperturbed autonomous systems for example [25], many of them have not been addressed yet and the field of developing methods for inferring dynamical properties from observations is largely open. In this thesis I address the methodology of inferring dynamical properties from observations of perturbed systems.

Three novel methods in total are presented in this thesis, each for a slightly different setting. The method of Chapter 2 was developed with spiky oscillators in mind and is therefore ideally suitable for inferring the PRC of coupled neurons, given that they are reasonably close to following limit cycle dynamics. But it is more general, it is suitable for any oscillator that we can collect time events from, whether that be because the oscillator is inherently discrete, or that we can only (or chose to only) observe certain time events (e.g. for continuous signals, events can simply mean threshold crossings). Therefore, this method is also applicable to discrete oscillators such as digital electronic components or potentially even more obscure ones such as multistable perception [139]. Its restriction is that the underlying dynamics of the studied system have to be well described with phase dynamics, which is usually the case only for weakly perturbed and not very noisy limit cycles. The method of Chapter 3 shares the same root idea with that of Chapter 2, but it is extended for inferring connectivity of oscillatory networks. Therefore, it is ideally suited for inferring the complete model connectivity (PRCs of all nodes and directed weighted coupling strengths) of small networks of neurons or other spiky cells, again, given that they roughly follow limit cycle dynamics. Its potential applications go beyond neuroscience however, with any network of oscillatory units being a potential candidate, e.g. interacting electronic oscillators or interacting biological rhythms (circadian rhythm, cardio-respiratory interactions etc.). It shares its restriction with the method of Chapter 2, namely that network nodes have to be decently described with phase dynamics. The method of Chapter 4 is most general both in the sense that it makes no assumptions on the dynamics, and that it can be easily extended to small networks. This makes it applicable to a range of systems and have adaptable goals, whether that be forecasting, quantifying system bifurcations or obtaining the isolated system model. But it comes with its own caveats, namely, it has a number of hyperparameters one must choose, and there is no systematic way to choose them. This means that inference can require a lot of trial and error, and if it does not work it is never certain whether it can inherently not be done of if one needs to try other hyperparameters.

When developing methods, one learns things about the nature of the problem that might not be obvious from the beginning. It is just as important to know which approaches lead to success as it is to know which approaches are bound to fail, so that others do not fall into the same pitfalls. An example of such a pitfall is the error of the phase derivative when neglecting the isochronal structure. Many inverse problems rely on estimating the phase and its derivative before inferring a phase equation. While the phase may be reasonably well estimated, its derivative is significantly less accurate without the knowledge of the isochronal structure, see Section 6.1.1 for more details. Our methods overcome this, as they do not rely on the assumption of radial isochrons.

Additionally, the process of method developing includes familiarizing oneself with the tools that are available, and possibly contributing to the development of new tools. For example, the methods of Chapters 2 and 3 use an iterative procedure to solve the nonlinear problem. Such an iterative procedure could be applied to many other nonlinear problems as well.

The methods presented in this thesis open up possibilities to study systems that were before out of reach, or at least to study some system from another point of view and in more detail. For example, in Chapter 5, I analyzed human gait with the method of Chapter 4 and found intriguing results, namely that human gait might not be chaotic. This shows how all these methods can easily transcend the realm of method studies and become interdisciplinary tools, tacking pressing problems in a range of scientific fields.

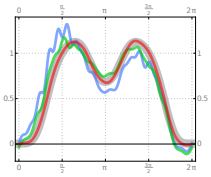
Additions

Software

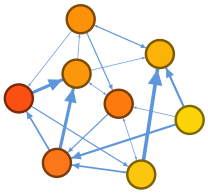
Throughout the course of my PhD I have written a lot of code. Here are a few codes that are well presented and complement my thesis.

OscillatorSnap
This is a high level Python package for inferring dynamics
using recurrent neural networks and machine learning.
It contains the examples presented in Chapter 4 [52].
It uses Keras [120] with TensorFlow [119]
for the lower level machine learning implementations.
Available at: `https://github.com/rokcestnik/oscillator_snap`
as well as on the Python Packaging Index PyPI as **oscillator_snap**.

Isochrons
This is a c code implementation of a backward integration algorithm
for computing isochrons from dynamical equations.
Figure 1.4 in the Introduction was generated with this code.
Available at: `https://github.com/rokcestnik/isochrons`.



Phase reponse curve inference
This a a c code implementation of
the inference method presented in Chapter 2 [50].
It uses the library Eigen [128] for the linear algebra.
Available at: `https://github.com/rokcestnik/prc_inference`.



Network inference
This is a c code implementation of
the network inference method presented in Chapter 3 [51].
It uses the library Eigen [128] for the linear algebra.
Available at: `https://github.com/rokcestnik/network_reconstruction`.

Cover page

The cover of this thesis was generated with the help of an oscillator:

$$\begin{aligned} \dot{x} &= \quad y - \sin(y)x \\ \dot{y} &= -x + \cos(x)y \end{aligned} \tag{A1}$$

which is the same oscillator as in Figure 1.4D. Its radially smallest limit cycle was rotated counter-clockwise by an angle of $i \times 2\pi/1000$ and then time-evolved for $T_0/10$ where T_0 is the natural period of the limit cycle. This process was repeated n times for each $i \in [0, 999]$ and the final "folded"[1] limit cycles were plotted transparently with yellow for $n = 8$, magenta for $n = 9$ and cyan for $n = 10$ with additive color mixing. Finally the color intensity was non-linearly rescaled making the image more vibrant.

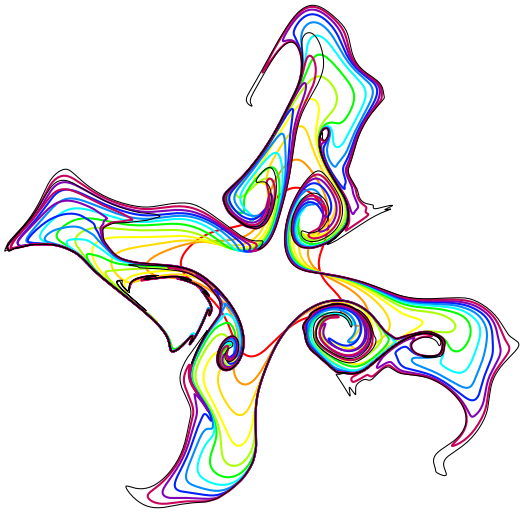


FIGURE A1: Folding induced by repeated rotation and time-evolution. The smallest limit cycle of system (A1) is plotted in red and then the following iterations of the folding process in colors following the rainbow through yellow, green, blue to purple. The final, 11^{th} iteration is depicted with a thin black line.

The backside cover of the thesis was generated by first evaluating the isochronal structure of the same oscillator (A1). This was accomplished with a code I made publically available, see *Isochrons* in the Software section. Then the structure was streched with the radial transformation:

$$r \to r\left[1 + \frac{1}{5 - r}\right] \tag{A2}$$

and the isochrones colored with a rainbow pallete.

[1] Iterated rotating and time evolving induces folding. An example of this can be seen in Figure A1 (smallest limit cycle of system (A1), repeatedly rotated for an angle of $2\pi/16$ and time-evolved for $T_0/16$).

Bibliography

[1] P. S. Landa. Examples of self-oscillatory systems with time delay. *Springer*, 1996.

[2] L. Glass and M. C. Mackey. Pathological conditions resulting from instabilities in physiological control systems. *Annals of the New York Academy of Sciences*, 316(1):214–235, 1979.

[3] S. H. Strogatz. Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering. *Westview Press*, 1994.

[4] O. E. Roessler. An equation for continuous chaos. *Physics Letters*, 57A(5):397–398, 1976.

[5] A. Pikovsky and J. Kurths. Coherence resonance in a noise-driven excitable system. *Physical Review Letters*, 78(5), 1997.

[6] E. M. Izhikevich and R. FitzHugh. Fitzhugh-nagumo model. *Scholarpedia*, 1(9):1349, 2006.

[7] K. Josic, E. T. Shea-Brown, and J. Moehlis. Isochron. *Scholarpedia*, 1(8):1361, 2006.

[8] J. T. C. Schwabedal, A. Pikovsky, B. Kralemann, and M. Rosenblum. Optimal phase description of chaotic oscillators. *Physical Review E*, 85(026216), 2012.

[9] J. T. C. Schwabedal and A. Pikovsky. Phase description of stochastic oscillations. *Physical Review Letters*, 110(204102), 2013.

[10] E. M. Izhikevich. Dynamical systems in neuroscience: The geometry of excitability and bursting. *The MIT Press*, 2007.

[11] M. Hénon. On the numerical computation of Poincaré maps. *Physica D*, 5:412–414, 1982.

[12] B. Kralemann, L. Cimponeriu, M. Rosenblum, A. Pikovsky, and R. Mrowka. Uncovering interaction of coupled oscillators from data. *Physical Review E*, 76(5), 2007.

[13] B. Kralemann, L. Cimponeriu, M. Rosenblum, A. Pikovsky, and R. Mrowka. Phase dynamics of coupled oscillators reconstructed from data. *Physical Review E*, 77(6), 2008.

[14] B. Kralemann, M. Frühwirth, A. Pikovsky, M. Rosenblum, T. Kenner, J. Schaefer, and M. Moser. In vivo cardiac phase response curve elucidates human respiratory heart rate variability. *Nature Communications*, 4:2418, 2013.

[15] A. Pikovsky, M. Rosenblum, and J. Kurths. Synchronization: A universal concept in nonlinear sciences. *Cambridge University Press*, 2001.

[16] A. T. Winfree. The geometry of biological time. *Springer*, 1980.

[17] S. Shirasaka, W. Kurebayashi, and H. Nakao. Phase-amplitude reduction of transient dynamics far from attractors for limit-cycling systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 27(2):023119, 2017.

[18] K. K. Lin, K. C. A. Wedgwood, S. Coombes, and L. Young. Limitations of perturbative techniques in the analysis of rhythms and oscillations. *Journal of Mathematical Biology*, 66:139–161, 2013.

[19] K. C. A. Wedgwood, K. K. Lin, R. Thul, and S. Coombes. Phase-amplitude descriptions of neural oscillator models. *The Journal of Mathematical Neuroscience*, 3(1):2, 2013.

[20] J. Rinzel. Discussion: Electrical excitability of cells, theory and experiment: Review of the hodgkin-huxley foundation and an update. *Bulletin of Mathematical Biology*, 52(1):3–23, 1990.

[21] R. R. Llins. Intrinsic electrical properties of mammalian neurons and cns function: a historical perspective. *Frontiers in Cellular Neuroscience*, 8:320, 2014.

[22] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Physical Review Letters*, 45:712–716, 1980.

[23] J. Justice. Analytic signal processing in music computation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(6):670684, 1979.

[24] D. Gabor. Theory of communication. *Institution of Electrical Engineering*, 93(3):429–457, 1946.

[25] F. Takens. Detecting strange attractors in turbulence. *Dynamical Systems and Turbulence*, 898:336–381, 1981.

[26] G. E. Uhlenbeck and L. S. Ornstein. On the theory of Brownian motion. *Physical Review*, 36:823–841, 1930.

[27] B. Pietras and A. Daffertshofer. Network dynamics of coupled oscillators and phase reduction techniques. *Physics Reports*, 2019.

[28] E. Ott and T. M. Antonsen. Low dimensional behavior of large systems of globally coupled oscillators. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 18(3):037113, 2008.

[29] J. Buck and E. Buck. Synchronous fireflies. *Scientific American*, 234(5):7485, 1976.

[30] Y. Fujino, B. M. Pacheco, S. Nakamura, and P. Warnitchai. Synchronization of human walking observed during lateral vibration of a congested pedestrian bridge. *Earthquake Engineering & Structural Dynamics*, 22(9):741758, 1993.

[31] P. Dallard, T. Fitzpatrick, A. Flint, A. Low, R. R. Smith, M. Willford, and M. Roche. London millennium bridge: Pedestrian-induced lateral vibration. *Journal of Bridge Engineering*, 6(6):412417, 2001.

[32] Y. Kuramoto. Self-entrainment of a population of coupled non-linear oscillators. *International Symposium on Mathematical Problems in Theoretical Physics Lecture Notes in Physics*, page 420422, 1975.

[33] K. McGoff, S. Mukherjee, and N. Pillai. Statistical inference for dynamical systems: A review. *Statistics Surveys*, 9:209252, 2015.

[34] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424, 1969.

[35] M. G. Leguia, C. G. B. Martnez, I. Malvestio, A. T. Campo, R. Rocamora, Z. Levnajić, and R. G. Andrzejak. Inferring directed networks using a rank-based connectivity measure. *Physical Review E*, 99(1), 2019.

[36] I. Malvestio, T. Kreuz, and R. G. Andrzejak. Robustness and versatility of a nonlinear interdependence method for directional coupling detection from spike trains. *Physical Review E*, 96(2), 2017.

[37] G. Sugihara, R. May, H. Ye, C.-H. Hsieh, E. Deyle, M. Fogarty, and S. Munch. Detecting causality in complex ecosystems. *Science*, 338(6106):496500, 2012.

[38] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115133, 1943.

[39] R. G. M. Morris. D.o. hebb: The organization of behavior, wiley: New york; 1949. *Brain Research Bulletin*, 50(5-6):437, 1999.

[40] P. J. Werbos. The roots of backpropagation: from ordered derivatives to neural networks and political forecasting. *J. Wiley and Sons*, 1994.

[41] G. E. Moore. Cramming more components onto integrated circuits. *Proceedings of the IEEE*, 86(1):8285, 1998.

[42] L. Deng and D. Yu. Deep learning: Methods and applications. *NOW Publishers*, (MSR-TR-2014-21), 2014.

[43] I. Goodfellow, Y. Bengio, and A. Courville. Deep learning. *MIT Press*, 2016.

[44] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354359, 2017.

[45] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556v6*, 2015.

[46] L. Deng, J. Li, J. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams, et al. Recent advances in deep learning for speech research at microsoft. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[47] D. Svozil, V. Kvasnicka, and J. Pospichal. Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems*, 39:43–62, 1997.

[48] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv:1406.1078*, 2014.

[49] I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. pages 1017–1024. Omnipress, 2011.

[50] R. Cestnik and M. Rosenblum. Inferring the phase response curve from observation of a continuously perturbed oscillator. *Scientific Reports*, 8(13606), 2018.

[51] R. Cestnik and M. Rosenblum. Reconstructing networks of pulse-coupled oscillators from spike trains. *Physical Review E*, 96(012209), 2017.

[52] R. Cestnik and M. Abel. Inferring the dynamics of oscillatory systems using recurrent neural networks. *Chaos*, 29(063128), 2019.

[53] L. Glass and M. C. Mackey. From clocks to chaos: The rhythms of life. *Princeton University Press*, 1988.

[54] J. Rinzel and B. Ermentrout. Analysis of neural excitability and oscillations. In C. Koch and I. Segev, editors, *Methods of Neuronal Modeling*, pages 251–292. MIT Press, Cambridge, 1998.

[55] C. C. Canavier. Phase response curve. *Scholarpedia*, 1(12):1332, 2006.

[56] E. Izhikevich. The dynamical systems in neuroscience: Geometry of excitability and bursting. *The MIT Press*, 2007.

[57] M. R. Guevara, A. Shrier, and L. Glass. Phase resetting of spontaneously beating embryonic ventricular heart cell aggregates. *American Journal of Physiology*, 251:1298–1305, 1986.

[58] T. Imai, K. Ota, and T. Aoyagi. Robust measurements of phase response curves realized via multicycle weighted spike-triggered averages. *Journal of the Physical Society of Japan*, 86:024009, 2017.

[59] G. B. Ermentrout, R. F. Galán, and N. N. Urban. Relating neural dynamics to neural coding. *Physical Review Letters*, 99:248103, 2007.

[60] K. Ota, M. Nomura, and T. Aoyagi. Weighted spike-triggered average of a fluctuating stimulus yielding the phase response curve. *Physical Review Letters*, 103:024101, 2009.

[61] K. Morinaga, R. Miyata, and T. Aonishi. Optimal colored noise for estimating phase response curves. *Journal of the Physical Society of Japan*, 84:094801, 2015.

[62] W. J. Freeman. Hilbert transform for brain waves. *Scholarpedia*, 2(1):1338, 2007.

[63] R. F. Galán, G. B. Ermentrout, and N. N. Urban. Efficient estimation of phase-resetting curves in real neurons and its significance for neural-network modeling. *Physical Review Letters*, 94:158101, 2005.

[64] D. Hansel, G. Mato, and C. Meunier. Synchrony in excitatory neural networks. *Neural Comput.*, 7:307–337, 1995.

[65] C. Morris and H Lecar. Voltage oscillations in the barnacle giant muscle fiber. *Biophysical Journal*, 35:193 – 213, 1981.

[66] J. Guckenheimer. Isochrons and phaseless sets. *Journal of Mathematical Biology*, 1:259–273, 1975.

[67] B. van der Pol. On relaxation-oscillations. *Philosophical Magazine*, 2(7):978–992, 1926.

[68] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Physical Review Letters*, 45:712–716, 1980.

[69] T. Sauer, J. A. Yorke, and M. Casdagli. Embedology. *Journal of Statistical Physics*, 65:579–616, 1991.

[70] B. Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of Computation*, 51:699–706, 1988.

[71] J. Teramae, H. Nakao, and G. B. Ermentrout. Stochastic phase reduction for a general class of noisy limit cycle oscillators. *Physical Review Letters*, 102:194102, 2009.

[72] G. N. Milstein. Approximate integration of stochastic differential equations. *Teoriya Veroyatnostei i ee Primeneniya*, 19:583–588, 1974.

[73] J. L. Hindmarsh and R. M. Rose. A model of neuronal bursting using three coupled first order differential equations. *Proceedings of the Royal Society of London B: Biological Sciences*, 221:87–102, 1984.

[74] O. Sporns, G. Tononi, and R. Kotter. The human connectome: A structural description of the human brain. *Computational Biology*, 1(4):245–251, 2005.

[75] R. Mrowka, L. Cimponeriu, A. Patzak, and M.G. Rosenblum. Directionality of coupling of physiological subsystems - age related changes of cardiorespiratory interaction during different sleep stages in babies. *American Journal of Physiology*, 145:R1395–R1401, 2003.

[76] G. Wang, P. Yang, X. Zhou, K. L. Swanson, and A. A. Tsonis. Directional influences on global temperature prediction. *Geophysical Research Letters*, 39:L13704, 2012.

[77] T. S. Gardner et al. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301:102–105, 2003.

[78] E. L. Berlow et al. Interaction strengths in food webs: issues and opportunities. *Journal of Animal Ecology*, 73:585–598, 2004.

[79] M. Timme. Revealing network connectivity from response dynamics. *Physical Review Letters*, 98:224101, 2007.

[80] A. Aertsen and G. L. Gerstein. Evaluation of neuronal connectivity: sensitivity of cross-correlation. *Brain Research*, 340:341–354, 1985.

[81] M. G. Rosenblum and A. S. Pikovsky. Detecting direction of coupling in interacting oscillators. *Physical Review E*, 64(10):045202, 2001.

[82] B. Kralemann, A. Pikovsky, and M. Rosenblum. Reconstructing phase dynamics of oscillator networks. *Chaos*, 21:025104, 2011.

[83] I. H. Stevenson et al. Inferring functional connections between neurons. *Current Opinion in Neurobiology*, 18:582–588, 2008.

[84] B. Kralemann, A. Pikovsky, and M. Rosenblum. Reconstructing effective phase connectivity of oscillator networks from observations. *New Journal of Physics*, 16:085013, 2014.

[85] W.D. Penny, V. Litvak, L. Fuentemilla, E. Duzel, and K.J. Friston. Dynamic Causal Models for phase coupling. *Journal of Neuroscience Methods*, 183(1):19–30, 2009.

[86] T. Stankovski, A. Duggento, P. V. E. McClintock, and A. Stefanovska. Inference of time-evolving coupled dynamical systems in the presence of noise. *Physical Review Letters*, 109:024101, 2012.

[87] R. Mirollo and S. Strogatz. Synchronization of pulse-coupled biological oscillators. *SIAM J. Appl. Math.*, 50:1645–1662, 1990.

[88] R. M. Memmesheimer and M. Timme. Designing the dynamics of spiking neural networks. *Physical Review Letters*, 97:188101, 2006.

[89] A Pikovsky. Reconstruction of a neural network from a time series of firing rates. *Physical Review E*, 93:062313, 2016.

[90] M. E. Marhic. Fiber optical parametric amplifiers, oscillators and related devices. *Cambridge University Press*, 2008.

[91] M. Tooley. Electronic circuits: Fundamentals and applications. *Newnes*, 2002.

[92] I. R. Epstein and J. A. Pojman. An intorduction to nonlinear chemical dynamics, oscillations, waves, patterns, and chaos. *Oxford University Press*, 1998.

[93] K. M. Stiefel and G. B. Ermentrout. Neurons as oscillators. *J Neurophysiol*, 116:2950 2960, 2016.

[94] A. Destexhe A. Babloyantz. Is the normal heart a periodic oscillator? *Biological Cybernetics*, 58(3):203–11, 1988.

[95] R. Foster and L. Kreitzman. Circadian rhythms: A very short introduction. *Oxford University Press*, 2017.

[96] G. Buzski. Rhythms of the brain. *Oxford University Press*, 2006.

[97] Henk A. Dijkstra. Nonlinear climate dynamics. *Cambridge University Press*, 2013.

[98] Henning U. Voss, Paul Kolodner, Markus Abel, and Jürgen Kurths. Amplitude equations from spatiotemporal binary-fluid convection data. *Physical Review Letters*, 83:3422–3425, 1999.

[99] H. Voss, M. J. Bünner, and M. Abel. Identification of continuous, spatiotemporal systems. *Physical Review E*, 57:2820–2823, 1998.

[100] M. Abel. Nonparametric modeling and spatiotemporal dynamical systems. *International Journal of Bifurcation and Chaos*, 14:2027–2039, 2004.

[101] H. Whitney. Differentiable manifolds. *Ann. Math.*, 37:645–680, 1936.

[102] J. Guckenheimer and P. Holmes. Nonlinear oscillations, dynamical systems, and bifurcations of vector fields. *Springer*, 2002.

[103] M. Quade, M. Abel, K. Shafi, R. K. Niven, and B. R. Noack. Prediction of dynamical systems by symbolic regression. *Physical Review E*, 94:012214, 2016.

[104] M. Schmidt and Lipson H. Distilling free-form natural laws from experimental data. *Science*, 324:81–85, 2009.

[105] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Physical Review Letters*, 120:024102, 2018.

[106] J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott. Using machine learning to replicate chaotic attractors and calculate lyapunov exponents from data. *Chaos*, 27(12):121102, 2017.

[107] Z. Lu, B. R. Hunt, and E. Ott. Attractor reconstruction by machine learning. *Chaos*, 29:061104, 2018.

[108] R. Zimmermann and U. Parlitz. Observing spatio-temporal dynamics of excitable media using reservoir computing. *Chaos*, 28:043118, 2018.

[109] Z. Chase Lipton. A critical review of recurrent neural networks for sequence learning. *arXiv:1506.00019*, 2015.

[110] H. Sak, A. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *INTERSPEECH-2014*, pages 338–342, 2014.

[111] A. Politi. Lyapunov exponent. *Scholarpedia*, 8(3):2722, 2013.

[112] B. A. Pearlmutter. Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1:263–269, 1989.

[113] K. Funahashi and Y. Nakamura. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6:801–806, 1993.

[114] C. A. Bailer-Jones, D. J. MacKay, and P. J. Withers. A recurrent neural networks for modelling dynamical systems. *Network: Computation in Neural Systems*, 9(4):531–47, 1998.

[115] R. Falahian, M. Mehdizadeh Dastjerdi, M. Molaie, S. Jafari, and S. Gharibyadeh. Artificial neural network-based modelling of brain response to flicker light. *Nonlinear Dynamics*, 81:1951–1967, 2015.

[116] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[117] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv:1409.1259*, 2014.

[118] C. Olah. Understanding LSTM networks. https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015.

[119] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. https://tensorflow.org, 2015.

[120] Keras team. Keras. https://github.com/keras-team/keras, 2019.

[121] J. Peters. Policy gradient methods. *Scholarpedia*, 5(11):3698, 2010.

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

[123] M. T. Rosenstein, J. J. Collins, and C. J. De Luca. A practical method for calculating largest lyapunovs exponents from small data sets. *Physica D*, 65:117–134, 1993.

[124] P. Kuchment. Floquet theory for partial differential equations. *Birkhäuser*, 60, 1993.

[125] R. Cestnik. Oscillator snap. https://github.com/rokcestnik/oscillator_snap, 2019.

[126] A. Mauroy, I. Mezi, and J. Moehlis. Isostables, isochrons, and koopman spectrum for the actionangle representation of stable fixed point dynamics. *Physica D*, 261:1930, 2013.

[127] A. S. Pikovsky, M. G. Rosenblum, G. V. Osipov, and J. Kurths. Phase synchronization of chaotic oscillators by external driving. *Phisica D*, (104):219–238, 1997.

[128] G. Guennebaud, B. Jacob, et al. Eigen v3. https://eigen.tuxfamily.org, 2010.

[129] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e918, 2015.

[130] J. K. Moore, S. K. Hnat, and A. J. van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. ZENODO https://zenodo.org/record/13030#.XqhrynUzY3E, 2014.

[131] B. L. Rankin, S. K. Buffo, and J. C. Dean. A neuromechanical strategy for mediolateral foot placement in walking humans. *Journal of Neurophysiology*, 112(2):374383, 2014.

[132] M. W. Whittle. Three-dimensional motion of the center of gravity of the body during walking. *Human Movement Science*, 16(2-3):347355, 1997.

[133] S. M. Bruijn, J. H. van Dien, O. G. Meijer, and P. J. Beek. Is slow walking more stable? *Journal of Biomechanics*, 42(10):15061512, 2009.

[134] S. M. Bruijn, J. H. van Dien, and A. Daffertshofer. Beta activity in the premotor cortex is increased during stabilized as compared to normal walking. *Frontiers in Human Neuroscience*, 9, 2015.

[135] S. M. Bruijn, O. G. Meijer, P. J. Beek, and J. H. van Dien. Assessing the stability of human locomotion: a review of current measures. *Journal of The Royal Society Interface*, 11(90):20130900, 2014.

[136] D. J. Miller, N. Stergiou, and M. J. Kurz. An improved surrogate method for detecting the presence of chaos in gait. *Journal of Biomechanics*, 39(15):28732876, 2006.

[137] Michael G. Rosenblum, Arkady S. Pikovsky, and Jürgen Kurths. Phase synchronization of chaotic oscillators. *Physical Review Letters*, 76(11):18041807, 1996.

[138] D. Wilson. Isostable reduction of oscillators with piecewise smooth dynamics and complex floquet multipliers. *Physical Review E*, 99(2), 2019.

[139] R. Moreno-Bote, J. Rinzel, and N. Rubin. Noise-induced alterations in an attractor network model of perceptual bistability. *Journal of Neurophysiology*, 98(3), 2007.

Acknowledgements

This work took several years to complete and along the way there were several people that helped make the endeavour easier, the quality finer and the process more meaningful:

- Michael Rosenblum, I was very fortunate to have had you as my supervisor. Thank you for your guidance and wisdom that can only come from years of scientific reflection and a person at peace with oneself and their surroundings.
- Andreas Daffertshofer, thank you for showing me the pros and cons of the academic lifestyle and using your network to help me work with interesting data. Thank you also for the help with the organization of the thesis, guiding me through administration and communicating with the committee.
- Arkady Pikovsky, thank you for giving me ample opportunity to improve and grow on my path to an academic career. Opportunities orchestrated to meet and talk to people did not go unnoticed. Thank you for firmly leading the COSMOS project.
- My defense committee: Arkady Pikovsky, Ralph Andrzejak, Olga Sosnovtseva, Klaus Lehnertz and Jaap van Dieën, thank you for investing time in reviewing my thesis and giving valuable feedback.
- Caroline Reid, thanks for the great attitude and bonding the COSMOS group together. I could not imagine a better person for coordinating our projects.
- Franzi Peter, thank you for your positive attitude and our runs through the fields.
- Gloria Checcini, thank you for your positive attitude sharing the office and many academics meetings.
- Janis Goldschmidt, thank you for your elaborations on everything and the programming/technical problems.
- Ralph, Erik, Erik, XunhMing, Chris, Olek, thank you for the calm and stable group environment in which one has a chance to prosper. Erik Teichmann, thank you for proofreading the thesis.
- Nicolas Deschle and Bastian Pietras, thank you for welcoming me in Amsterdam and treading the paths before me.
- Moira, Sabrina, Nick and other VU students, thank you for introducing me to the good part of the Dutch culture. Nick, thanks for the tea.
- The people of COSMOS, thank you for great company and contributing to the great atmosphere of the COSMOS project.
- Ambrosys, Markus, Markus, Thomas, Franz, Tilo, Greta, thank you for the experience of a young industry environment, your programming skills were something for me to look up to and helped me improve.
- The many people I met and discussed with on conferences, Ivan, Verena, Yulia, Grant[2], Justus, Marcelo, Christina, and others, thank you for a good atmosphere during conferences.
- Sissi, thank you mi amor for your love and support throughout good and bad. You made the years of my PhD brighter.
- Sonja, Bojan, Petra and family, hvala za neomajno podporo ki sem jo čutil neglede na razdaljo.

Summary

Oscillatory processes are omnipresent in the animate and inanimate world. Oscillations come in different forms, be that purely periodic, largely recurrent such as low-dimensional chaotic systems, or more erratic such as high dimensional systems and systems under impact of random noise. They can be observed across (almost) all temporal and spatial scales. Take the example of physiological systems. Rhythms are present on the microscopic spatial scale in cells and their compounds, on the mesoscopic scale in groups of cells like populations of neurons and other tissues, and on the macroscopic scale in organs like the brain, the heart, lungs, etc., up to even rhythms spanning entire organisms. Prime examples of these oscillations are neuronal spiking, beating of the heart and the circadian rhythm, together spanning over five orders of magnitude of both spacial and temporal scales. This abundance of oscillatory processes calls for in-depth theorizing of their emergence, which typically relies on studying oscillators in operation. The ultimate goal of such an endeavor is to identify generic quantities that help classify the dynamics of oscillators. A corresponding mathematical treatment that may eventually allow for predicting oscillatory behavior requires first and foremost an acceptable simplification of the oscillatory process under study.

A natural language for the dynamics of oscillators is phase dynamics, which implies that an oscillator can be parameterized by a single, real-valued state variable - the phase - which is typically imagined as evolving on a circle and hence bounded to the interval from 0 to 2π. Within the scope of phase dynamics the defining property of an oscillator is its so-called phase response curve. This curve describes the sensitivity of the phase to external perturbations and is utilized in many theoretical and experimental studies. It may be estimated experimentally, in which the system under study is isolated and subjected to particularly designed and controlled stimuli. Unfortunately, isolation and control of a system are often not feasible due to many reasons. For instance, if the system under study is simply too large to be controlled, or if stimulating it could potentially cause damage or is otherwise irreversible - in vivo assessments of nervous activity might be a good example for this dilemma. In these cases, any quantification has to rely on the normal, ongoing state of system behavior, i.e. by observing the system in its natural environment, under free-running conditions. In this thesis, I present different approaches to this inverse problem, concentrating particularly on phase dynamics and the phase response curve.

Chapter 1 provides a general overview into the field of oscillatory

dynamics, in which I categorize the different types of oscillators and give an introduction to phase dynamics. This is followed by a brief outline of techniques used for obtaining dynamical properties from observations. These range from signal embedding to state-of-the-art machine learning. Throughout the thesis, I illustrate my numerical methods by applying them to seminal oscillatory models, thus showcasing their strengths and weaknesses as well as applicability to real data. The development of my methods was motivated by the long term goal of untangling the dynamics of many real world systems, notably the dynamics of biological neural networks.

In Chapter 2, I present a method for inferring the phase response curve from observations of a forced oscillator. It assumes the oscillator has a stable limit cycle and the forcing is weak enough so that the system can be well described with phase dynamics. The forcing may be stochastic or coming from other dynamical systems so long as it is varied enough to ensure the signal of the observed oscillator is not periodic. Both the signal of the oscillator, as well as the forcing, have to be observed to perform the inference. The method expresses the optimization problem in terms of the integrated Winfree phase form, where the phase response curve's Fourier coefficients are iteratively updated to better fit the phase model to the data. I also introduce an error measure to help monitor the convergence of the solution. This measure can be computed solely from observations without any additional knowledge of the system, allowing to monitor the quality of inference even with data where the ground truth is unknown beforehand. Results of the inference are illustrated for two common oscillatory models, the Van der Pol oscillator and the Morris-Lecar neuron, both under impact of weak random perturbations. While the first one is more mathematically motivated, the latter falls in the category of seminal neural models as it is a reduced excitation model applicable to systems having two non-inactivating voltage-sensitive conductances (originally it employed an instantaneously responding calcium conductance for excitation and a delayed potassium conductance for recovery in muscle fibers of the giant barnacle). Since my algorithm provides an error measure indicating the quality of the inference that can be calculated solely from data, and because it provides very promising results even when observations only include a few hundred oscillations, it appears very suitable for studying empirically observed oscillators. Note that in this study the perturbations were realized as random dynamic noise that can be considered omnipresent in experimental data. Hence, my approach is immediately applicable to inferring phase responses using ongoing activity.

This success motivated the subsequent study summarized in Chapter 3, where I generalized this approach in order to reconstruct a whole network of pulse-coupled oscillatory networks. In addition to phase dynamics I assumed that phases are instantaneously reset by incoming pulses. I employed two model systems: a network of toy oscillator models that generate 'spikes' as soon as their phase reaches a predefined threshold, and a more realistic network of Morris-Lecar neurons. From observing the spike trains of all units the method recovers the properties of all nodes (phase response curve and natural frequency) as well as the strength of all directed connections. As in the previous method, a few hundred observed pulses from each node suffice for a reliable reconstruction of a small network. That number does increase for larger networks and there is a hard lower limit of having to observe at least as many spikes as there are nodes.

The approach of Chapters 2 and 3, relies on reformulating the problem of inference onto a known optimization scheme. In particular, I used the least squares optimization, but others would work just as well. The novel feature comes in the combination with an iterative procedure. This iteration comes with an amazing convergence that can be readily monitored with my aforementioned error measure. As illustrated, the computational demand of both methods is acceptable and the amount of input data not excessively large. That is, both methods can be considered very efficient, both in terms of computational and data requirements.

In the subsequent Chapter 4, I explored an alternative approach of inferring dynamical properties from experimental observations. The reason for this was that the previous approaches, albeit successful, required perturbations to be weak and the system under study to be close to limit cycle behavior. In this alternative approach these conditions are not required. The approach consists of first selecting a generic model capable of reproducing the empirical observations. This model is then optimized to best fit the data, and therefore if its architecture allows it, it should obtain similar characteristics to the original system. One can then simply perform an experiment on the model as if experimenting on the original system. Such active control in experiments allows simple, well known practical methods to be utilized, thus putting within reach a wide range of dynamical quantities that are typically difficult to distill from data: the phase response curve and various scaling exponents just to name a few. While this scheme may seem straightforward, it comes with its own challenges. The most obvious one being how to infer the model in the first place. To tackle this, I employed artificial recurrent neural networks in combination with a mathematically robust machine

learning optimization algorithm referred to as stochastic gradient descent. This machine learning approach can successfully reproduce an array of empirical oscillatory signals as demonstrated in this chapter on several examples: chaotic Roessler oscillator, time delay induced oscillations from the Mackey-Glass equation, and even an excitable neuronal model FitzHugh-Nagumo. This is crucial to the entire approach since if the generic model does not fit the data well, then any property we obtain from it will be spurious. In contrast to the approach from Chapters 2 & 3, this turns out to be computationally costly and requires an appropriate choice of hyperparameters, like the number of nodes, the (type of) activation function, and so on. When it comes to analyzing experimental data, the latter architectural parameterization might not be obvious at all and may hence require repeated applications with trial and error. Yet I could conclude that recurrent neural networks have significant capacities for identifying a system's dynamics and are hence a great candidate for the role of a generic model in this approach. The predictive power of this machine learning approach is highly valuable and future studies will certainly continue following the route that I outlined here.

As a first step in this direction, I applied the machine learning approach to analyze experimental data obtained from perturbed human gait in Chapter 5. Participants walked on a treadmill and 3D kinematics were recorded. Gait was weakly perturbed by randomly changing the speed of the treadmill. In this pilot assessment I investigated the 3D velocity of the pelvis; the pelvis motion is often considered a marker of gait stability. The recurrent neural network model was trained using the corresponding time series and revealed that the pelvis' velocity followed stable limit-cycle dynamics rather than more complicated, chaotic behavior as suggested in several previous studies. Yet, these results should be taken with caution as they were based on a single subject assessment. Here, this analysis served as a mere proof-of-concept underscoring this usefulness of the machine learning approach advocated in Chapter 4 for investigating experimentally relevant data.

In the final Chapter 6, I discuss the applicability of the methods presented in this thesis. In particular the methods in Chapters 2 & 3 come with certain assumptions. I provide ideas about their outcome if these are not fulfilled. The approach in Chapters 4 & 5 requires a minimal amount of input data, which is a well-known problem in the machine-learning community. I relate this to the general problem of limited information content in data and use this link to elaborate on the more principal choices that have to be made when designing and tailoring methods for inferring dynamical properties from experimental data.