

This content has been downloaded from IOPscience. Please scroll down to see the full text.

Download details:

IP Address: 3.139.101.198

This content was downloaded on 26/04/2024 at 16:34

Please note that [terms and conditions apply](#).

You may also like:

[Aptasensors for Point-of-Care Diagnostics of Cancer](#)

[INVERSE PROBLEMS NEWSLETTER](#)

[INVERSE PROBLEMS NEWSLETTER](#)

[INVERSE PROBLEMS NEWSLETTER](#)

[Doppler Ultrasound: Physics, Instrumentation and Signal Processing \(Second Edition\)](#)

M Halliwell

# Nonlinear Dynamics

A hands-on introductory survey

Marc R Rousset

---

## Chapter 1

### Introduction

#### 1.1 What is a dynamical system?

Roughly speaking, a dynamical system is a system that evolves in time. More precisely, a dynamical system has the following three components:

- Some variable that acts like time, i.e. an independent variable that increases monotonically and independently of the evolution of the system, with other variables being indexed by our time variable.
- Some variables that describe the state of a system. These variables define a **state space**.
- A rule according to which the state evolves in time. We can think of the time evolution either as a time-dependent state, say  $x(t)$ , or as a **trajectory** in the state space.

The evolution rule can be encoded in any number of ways. Typically, we would have some sort of equation that tells us how to calculate future states given the current state. For example, we could have a **map**

$$x_{n+1} = f(x_n).$$

In this case, the ‘time’ variable is the step number  $n$ . The function  $f$  tells us how to calculate  $x$  at the next step of the evolution. All we need is an initial value of  $x$ , conventionally denoted  $x_0$ , to get this started.

We could also have a differential equation, such as Newton’s second law:

$$F(x) = m \frac{d^2x}{dt^2}.$$

Given a specification of the position-dependent force (e.g. Hooke’s law,  $F(x) = -kx$ ) and an initial condition (in this case an initial position and velocity), we can solve this equation for the position as a function of time.

If you think about it for a minute, you will find that a lot of things can be described as dynamical systems:

- Any system that can be described by differential equations. This includes mechanical systems, electrical systems, and chemical systems. Biological populations have often been modeled using differential equations as well.
- Maps show up as models of populations with discrete generations.
- In computer science, a Turing machine is a very formal description of a type of computer executing a stored program. The program is executed step-by-step, so this process can be thought of as a dynamical system.
- Nothing in the definition of a dynamical system says that the evolution has to be deterministic, so there are also stochastic dynamical systems, i.e. systems whose time evolution has a random component. Random environmental influences can be incorporated into population models; models of microscopic chemical systems often consider fluctuations due to the discrete nature of molecules; and so on. The theory of stochastic dynamical systems is fascinating, but in this book, we will focus on deterministic systems.

In mathematics, it is often useful to add a layer of abstraction in order to be able to treat many types of systems using a single framework. In the case of dynamical systems, the idea of time evolution is represented abstractly by a time evolution operator, here denoted  $\varphi^t$ . If we have a dynamical system with a state  $x(t)$ , then the time evolution operator has the following properties:

- $\varphi^t x(0) = x(t)$ . This just says that the time evolution operator generates a solution of the equations that govern the dynamical system given the initial condition.
- As a consequence, we must have  $\varphi^0$  as the identity operator. In other words,  $\varphi^0 x = x$ .
- $\varphi^t x(s) = x(t + s)$ . The time evolution operator works just as well on some time point  $s$  as it does on the initial condition.
- The latter implies that  $\varphi^{t+s} = \varphi^t \circ \varphi^s = \varphi^s \circ \varphi^t$ , where  $\circ$  represents operator composition. If you prefer, we can write this relationship as  $\varphi^{t+s} x = \varphi^t(\varphi^s x) = \varphi^s(\varphi^t x)$ .
- Note that  $\varphi^t \circ \varphi^s = \varphi^s \circ \varphi^t$  means that the time evolution operator commutes with itself.

In many cases, we don't need an explicit form for the time evolution operator. In fact, we usually don't. It's sometimes enough to know that an operator with the above properties exists.

In some cases, we can reconstruct the initial condition from the solution at some time  $t$  by running time backwards. In other words, the time evolution operator is defined both for positive and negative values of  $t$ . In more formal language,  $\varphi^t$  is an invertible operator. In many interesting cases, the time evolution operator is not invertible.

**Example 1.1.** The map  $x_{n+1} = x_n(1 - x_n)$  defined on  $0 \leq x \leq 1$  is not invertible since both  $x_n = a$  and  $x_n = 1 - a$  map to the same value of  $x_{n+1}$ .

In this book, you will be introduced to a variety of types of dynamical systems. It is hoped that you will begin to appreciate the similarities between different types of dynamical systems, as well as some of the conceptual similarities in the types of questions we might ask about them and in the methods used to answer those questions. We often want to know what kinds of solutions a given dynamical system might have (equilibrium points, oscillatory solutions, etc), and whether these solutions are stable or not. We also want to be able to map out the conditions under which certain types of solutions will be observed. And if we can't answer these questions analytically, we can study them numerically.

## 1.2 The law of mass action

I'm a chemist, so many of my examples come out of this field. In terms of learning nonlinear dynamics, it's neither here nor there where our examples come from. However, in the interests of us having a common language, let me present here the fundamental law of chemical kinetics, the law of mass action.

A chemist writing down a chemical reaction might mean one of two things:

- A simple statement about stoichiometry. When we write  $\text{H}_2 + \frac{1}{2}\text{O}_2 \rightarrow \text{H}_2\text{O}$ , all we're doing is stating the proportions of hydrogen and oxygen that react to make one equivalent of water. We are saying nothing about how this reaction happens.
- A process that occurs exactly as written is called an **elementary reaction**. For example, one of the elementary reactions in the hydrogen/oxygen reaction is  $\text{H}_2 + \text{O} \rightarrow \text{H} + \text{OH}$ . When we write this, we quite literally mean that a hydrogen molecule meets an oxygen atom, resulting in the transfer of a hydrogen atom to the oxygen.

A **reaction mechanism** is a list of all the elementary reactions that result in a particular overall chemical change.

The law of mass action says that the rate of an elementary reaction is proportional to the product of the concentrations of the reactants. The proportionality constant is called a **rate constant**, denoted by a lower-case  $k$ , usually with a subscript to designate a particular elementary reaction, since a mechanism will generally consist of many elementary reactions. Thus, the rate of the  $\text{H}_2 + \text{O}$  reaction shown above can be written  $k_f[\text{H}_2][\text{O}]$ , where the square brackets denote the concentrations<sup>1</sup> of the corresponding chemical species. Note that the rate of this elementary reaction does *not* depend on the concentrations of the products. However, because elementary reactions are always reversible (due to another law, the **law of microscopic**

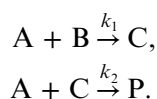
---

<sup>1</sup> Concentration is the amount of substance per unit volume. It can take many units, but in chemical kinetics, we would often express concentration in  $\text{mol L}^{-1}$ .

**reversibility**), the reverse reaction is also an elementary reaction, with rate  $k_r[\text{H}][\text{OH}]$ . (The subscripts  $f$  and  $r$  on the rate constants were chosen to denote the forward and reverse reactions. These directions are of course perfectly arbitrary.) Note also that each elementary reaction has its own rate constant, and this is true even when two elementary reactions are the reverse of each other. It is often the case that one direction of a reaction under some particular experimental conditions of interest is sufficiently slow to be negligible, so reactions will not always be treated as reversible in models.

The rate of change of a particular concentration is the sum of the rates of change due to all elementary reactions in which the corresponding species participates. This is perhaps best illustrated with an example.

**Example 1.2.** Consider the mechanism



Note the placement of rate constants over arrows to associate rate constants to reactions. The rate of change of  $[\text{A}]$  implied by this mechanism is

$$\frac{d[\text{A}]}{dt} = -k_1[\text{A}][\text{B}] - k_2[\text{A}][\text{C}].$$

The negative signs arise because A is removed by both reactions. Compare the rate equation for  $[\text{C}]$ :

$$\frac{d[\text{C}]}{dt} = k_1[\text{A}][\text{B}] - k_2[\text{A}][\text{C}].$$

## 1.3 Software

We can do some things explicitly with pen and paper, but for systems of even moderate complexity, we often can't get very far that way in nonlinear dynamics. Computers are often needed to assist in both analytical and numerical calculations. In my opinion, to learn nonlinear dynamics, you actually need three kinds of software:

**Specialized nonlinear dynamics simulation/analysis software:** I chose XPPAUT as the nonlinear dynamics software to be used in this book. Despite its age, it's still one of the more straightforward packages available with as wide a range of capabilities. Moreover, it's free, and it runs on a variety of computers. Apart from the documentation that comes with XPPAUT, you may also want to check out Bard Ermentrout's tutorial-style book [1], which walks users through many common calculations.

**A general-purpose numerical computing environment:** For general numerical calculations, I like `MATLAB` (or its free cousin, `OCTAVE`). `MATLAB` is heavily used across the sciences, so it's well worth learning if you haven't already. There are some differences between `MATLAB` and `OCTAVE` syntax, as well as some differences between their libraries. All of the code in this book has been written in the common subset of the two languages and will run in either environment.

**A symbolic algebra system:** Some analytical calculations are just too tedious to do by hand. In these cases, it pays to have access to a symbolic algebra system. At my home university, we use `MAPLE`, so that's what I used in this book.

My intent is for this to be a hands-on introduction to nonlinear dynamics, where you can try out the calculations in the book yourself, and then try to apply the same techniques to other problems. You will therefore find quite detailed instructions and sample code for using `XPPAUT`, `MATLAB/OCTAVE` and `MAPLE` throughout the book. If you use different software, then hopefully you will be able to adapt the procedures described in this book to the software that is available to you. Aside from the symbolic algebra system, this need not be an issue since `XPPAUT` and `OCTAVE` are free software. Installation instructions for `XPPAUT` and `OCTAVE` are provided in [appendix A](#).

## Reference

- [1] Ermentrout B 2002 *Simulating, Analyzing, and Animating Dynamical Systems* (Philadelphia: SIAM)