1st Latin American Summer School

Prof Jacob Bedrossian
University of Maryland, College Park USA

June 20, 2019

Contents

1 Introduction

These notes were devised for a week long intensive introduction to some topics in using partial differential equations for mathematical modeling. To fix examples which are accessible without a lot of background in the specific application topic, the kind of mathematical models we are discussing mostly apply to biology and chemistry, i.e. population dynamics of organisms or a system of multiple chemical species undergoing chemical reactions. In the last section, we briefly discuss the extension of some of the ideas to modeling the dynamics of fluids. These notes serve as a reference for the course – maybe not every topic and every example discussed in these notes will be covered in the course due to time constraints.

It is assumed the reader has some basic knowledge of vector calculus and linear algebra, but there is an appendix briefly reviewing some of the concepts we are using (also sets the notation being used).

2 Mathematical modeling and ODEs

In this section I will introduce some of the basic ideas behind using ordinary differential equations (ODEs) to model things in the real world. For illustrative purposes, I will mostly use examples from biology and chemistry, rather than physics, simply because there are some good examples that are accessible and interesting without requiring a lot of technical background in the field of study.

2.1 ODEs in dimension one for population dynamics

2.1.1 Exponential growth model

Consider the population of bacteria in a petri dish represented by $N(t)$, i.e. the number of bacteria as a function of time. Suppose that there are lots of available resources so that λr of the bacteria divides into two new bacteria every r hours. We then have the following relation:

$$N(t+r) = N(t) + \lambda r N(t). \tag{2.1}$$

In order to figure out the population of bacteria, you would need also to know how many you had at the beginning of the experiment – the so-called *initial condition*. This gives the *mathematical model* for the number of bacteria in the dish

$$N(t+r) = (1+\lambda r)N(t). \tag{2.2}$$
$$N(0) = N_0. \tag{2.3}$$

We can solves this and we have for all $J > 0$,

$$N(Jr) = (1+\lambda r)^J N_0. \tag{2.4}$$

There are so many bacteria, it seems silly to try to keep track of each individual one. Moreover, it doesn't really make sense to expect all the bacteria to be dividing on the same discrete set of times, this is of course not how bacteria divide. We should come up with a mathematical model for which t and N are continuous variables. Re-writing

$$\frac{N(t+r) - N(t)}{r} = \lambda N(t). \tag{2.5}$$

We want to think about sending $r \to 0$, so taking this limit gives

$$\frac{d}{dt}N(t) = \lambda N(t) \tag{2.6a}$$
$$N(0) = N_0. \tag{2.6b}$$

Theorem 2.1. *The solution to* (2.6) *is given by*

$$N(t) = N_0 e^{\lambda t}. \tag{2.7}$$

Proof. Write

$$\frac{1}{N(t)}\frac{d}{dt}N(t) = \lambda, \tag{2.8}$$

and then the backwards chain rule:

$$\frac{d}{dt}\log N(t) = 2\lambda, \tag{2.9}$$

Then integrate

$$\int_0^T \frac{d}{dt}\log N(t)dt = 2\lambda T \tag{2.10}$$

which gives

$$\log N(T) - \log N(0) = 2\lambda T, \tag{2.11}$$

and rearranging then gives (switching T back to t also),

$$N(t) = N_0 e^{\lambda t}. \tag{2.12}$$

□

2.1.2 Logistic equation

Example 1. E. Coli replicate at a very fast rate. Given enough resources, they can replicate in under around 30 minutes. Since we are measuring t in hours, this suggests that $\lambda = 2$ is a rough guess for how big we can take λ. Suppose that on the first day, you have a tiny culture of 100 E. Coli, i.e. $N_0 = 100$. Our mathematical model predicts that in 2 days, we should have around

$$N(48) = 100e^{2*48} \approx 4.9 \times 10^{43} \text{ E. Coli.} \tag{2.13}$$

For comparison, an E. Coli cell has a mass of around 10^{-15} kg, and so our mathematical model predicts you have around 10^{28} kg worth of E. Coli. All of these rates and numbers were very approximate, but given that the Earth has a mass of around 6×10^{24} kg, we can be pretty confident that something went horribly wrong. Everything was pretty hand-wavy there, but we're looking at a prediction that is astronomically inaccurate[1].

[1]This calculation is really only *very* sensitive to λ and the length of time you run the experiment. For example, if you change $\lambda = 2$ to $\lambda = 1$, you get about 70×10^6 kg of E. Coli by day 2. This is still ridiculous, but its nowhere the size of planets.

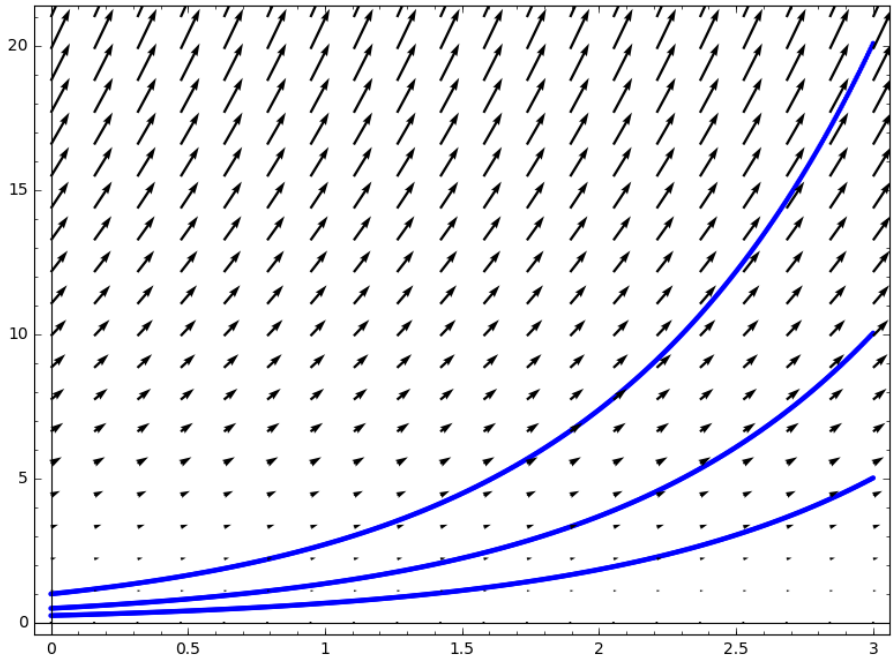

Figure 1: A plot of the solution to (2.6) for various initial conditions.

The key point in the above example was *"Given enough resources..."*. The original model (2.6) should only work well when the resources and space available to the bacteria can be assumed unlimited. Once the bacteria are numerous enough, they will start running out of resources (or space), which means that the original model (2.6) will no longer be a good approximation. It makes sense to replace the growth rate λ with something that decreases as the number of bacteria increases. Assuming this dependence is linear, we would get a growth rate $\lambda(1 - \frac{1}{A}N(t))$, and the model we have is the *logistic model*:

$$\frac{d}{dt}N(t) = \lambda\left(1 - \frac{1}{A}N(t)\right)N(t) \tag{2.14a}$$

$$N(0) = N_0. \tag{2.14b}$$

Note that when $\frac{1}{A}N(t)$ is very small, the growth rate is close to λ, so that the model is very similar to (2.6). When $N(t) > A$ the population is actually decreasing, i.e. the bacteria are dying due to lack of resources. When $N(t) = A$ the population is not changing - we call this *equilibrium*. This is one of the few nonlinear ODEs that can be solved without the aid of a computer, if you know a few tricks.

Theorem 2.2. *The solution to* (2.14a) *is given by*

$$N(t) = \frac{e^{\lambda t}N_0}{1 - \frac{N_0}{A} + \frac{1}{A}e^{\lambda t}N_0}. \tag{2.15}$$

Notice that, if $N_0 = A$, then $N(t) = A$ and that if $N_0 = 0$ then $N(t) = 0$. These are the two equilibria of the model. If $0 < N_0$, note that $\lim_{t\to\infty} N(t) = A$.

Proof. As before, re-write as:

$$\frac{1}{\left(1 - \frac{1}{A}N(t)\right)N(t)}\frac{d}{dt}N(t) = \lambda. \tag{2.16}$$

Then we use partial fractions

$$\frac{1}{\left(1-\frac{1}{A}N(t)\right)N(t)} = \frac{1}{N(t)} + \frac{1}{A\left(1-\frac{1}{A}N(t)\right)} \tag{2.17}$$

So,

$$\frac{d}{dt}\log N(t) - \frac{d}{dt}\log\left(1-\frac{N(t)}{A}\right) = \lambda \tag{2.18}$$

$$\frac{d}{dt}\log\frac{N(t)}{\left(1-\frac{N(t)}{A}\right)} = \lambda. \tag{2.19}$$

Integrating as above [] then gives

$$\log\frac{N(t)}{\left(1-\frac{N(t)}{A}\right)} - \log\frac{N_0}{\left(1-\frac{N_0}{A}\right)} = \lambda t. \tag{2.20}$$

Then,

$$\frac{N(t)}{\left(1-\frac{N(t)}{A}\right)} = e^{\lambda t}\frac{N_0}{\left(1-\frac{N_0}{A}\right)}, \tag{2.21}$$

which re-arranges to

$$N(t) = \frac{e^{\lambda t}\frac{N_0}{\left(1-\frac{N_0}{A}\right)}}{1+\frac{1}{A}e^{\lambda t}\frac{N_0}{\left(1-\frac{N_0}{A}\right)}} \tag{2.22}$$

$$= \frac{e^{\lambda t}N_0}{1-\frac{N_0}{A}+\frac{1}{A}e^{\lambda t}N_0}. \tag{2.23}$$

□

2.2 Method of linearization

Let's go back and consider this model:

$$\frac{d}{dt}N(t) = \lambda\left(1-\frac{1}{A}N(t)\right)N(t) \tag{2.24}$$

$$N(0) = N_0. \tag{2.25}$$

We expect that for small population sizes, the exponential growth model is actually fine. Let's see if that's true. Suppose that $N(t) = \epsilon f(t)$ and $N_0 = \varepsilon f_0$. This would give

$$\frac{d}{dt}f(t) = \lambda f(t) - \varepsilon\frac{\lambda}{A}f^2(t) \tag{2.26}$$

$$f(0) = f_0. \tag{2.27}$$

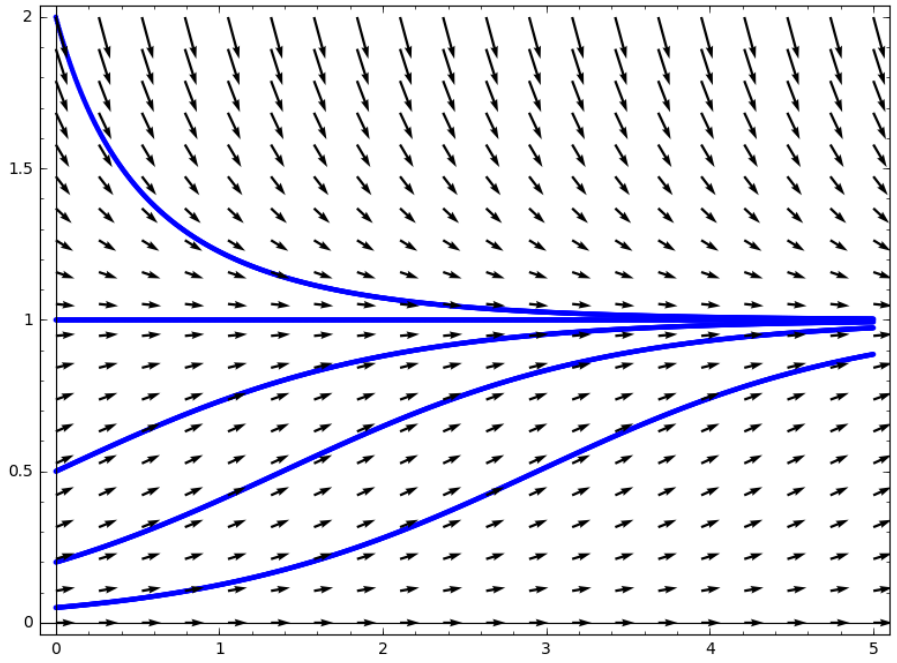

Figure 2: A plot of the solution to (2.14a) for various initial conditions.

I haven't actually changed the model, I just basically changed the units of N. However, suppose that $\varepsilon \ll \frac{\lambda}{A}$, i.e. much, much smaller. Then, as long as f wasn't too large, we could safely drop the last term and get left with

$$\frac{d}{dt}f(t) = \lambda f(t) \tag{2.28}$$

$$f(0) = f_0. \tag{2.29}$$

The exponential growth model! This shows that small population sizes will tend to grow exponentially.

Now, let's see if we can find an approximate model near the other equilibrium, $N(t) = A$. Hence, suppose that $N(t) = A + \varepsilon f(t)$. We then have

$$\frac{d}{dt}f(t) = -\lambda f(t) - \varepsilon\lambda f^2(t) \approx -\lambda f(t) \tag{2.30}$$

$$N(0) = f_0. \tag{2.31}$$

This model is an exponential decay model

$$f(t) = f_0 e^{-\lambda t}. \tag{2.32}$$

This shows that populations close to equilibrium $N(t) = A$ will converge to this equilibrium exponentially fast.

What we're seeing is the *method of linearization*, in which near equilibria at least, we approximate the nonlinear dynamics by linear dynamics. This is sort of analogous to when one locally approximates a nonlinear curve by its tangent line. Let's see this method in action. Suppose that

we had a general model

$$\frac{d}{dt}N(t) = G(N(t)) \tag{2.33}$$
$$N(0) = N_0 \tag{2.34}$$

and that you have an equilibrium point $G(A) = 0$. Doing the same procedure by considering $N(t) = A + \varepsilon f(t)$ and applying Taylor's theorem gives:

$$\varepsilon\frac{d}{dt}f(t) = G(A + \varepsilon f(t)) = G(A) + \varepsilon G'(A)f(t) + \frac{\varepsilon^2}{2}G''(A)f^2(t) + ... \tag{2.35}$$

Therefore, if we use $G(A) = 0$, divide by ε and then drop all the terms with at least one ε, we get the *linearized equation*

$$\frac{d}{dt}f(t) = G'(A)f(t) \tag{2.36}$$
$$f(0) = f_0 \tag{2.37}$$

which is then solved by

$$f(t) = f_0 e^{G'(A)t}. \tag{2.38}$$

Therefore, if $G'(A) < 0$, $f(t)$, i.e. the small perturbation, converges exponentially back to A. In this case, we say the equilibrium is *stable*. If $G'(A) > 0$, then $f(t)$ grows exponentially. In this case, we say the equilibrium is *unstable*. If $G'(A) = 0$, then the equilibrium is *neutral* and a higher order expansion is needed to make a good guess about stability.

2.3 2x2 systems for population dynamics

Many systems of interest in the real-world have more than one variable. In fact, many of them can have hundreds, thousands, or even millions. Let's start with the case of just two variables, which is already representative of some of the new aspects one sees when going from one unknown to many. A very simple and beautiful model for two species interacting in the same environment is the *Lotka-Volterra model*. Suppose one has two populations: a population of rabbits $u(t)$ and a population of foxes $v(t)$ (prey and predator). The rabbits we assume have essentially unlimited resources from the available plant food sources and grow naturally at a rate a, but are consumed at some rate by the foxes b. The Foxes on the other hand, will die in the absence of rabbits at a rate of d and can only grow by consuming the rabbits at a rate c. This gives

$$\frac{d}{dt}\begin{pmatrix}u(t)\\v(t)\end{pmatrix} = \begin{pmatrix}(a - bv(t))u(t)\\(cu(t) - d)v(t)\end{pmatrix}. \tag{2.39a}$$
$$\begin{pmatrix}u(0)\\u(0)\end{pmatrix} = \begin{pmatrix}u_0\\u_0\end{pmatrix}. \tag{2.39b}$$

To look for equilibria states for this system, we look for where the time-derivative vanishes:

$$\begin{pmatrix}0\\0\end{pmatrix} = \begin{pmatrix}(a - bv(t))u(t)\\(cu(t) - d)v(t)\end{pmatrix}. \tag{2.40}$$
$$\tag{2.41}$$

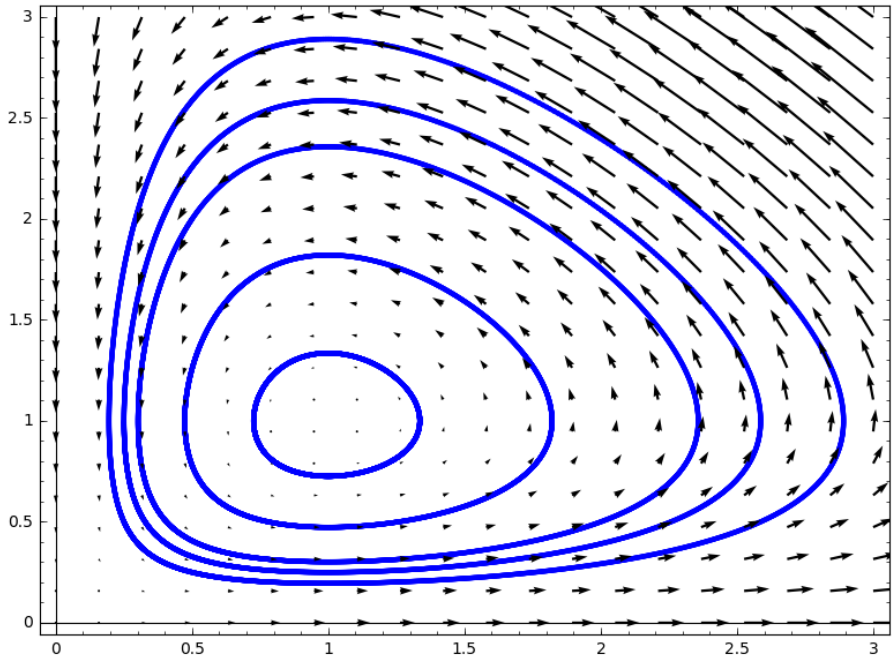

Figure 3: Solutions to (2.45a) for various initial conditions.

There are two solutions:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{d}{c} \\ \frac{a}{b} \end{pmatrix}. \tag{2.42}$$

The first is the "trivial" equilibria where both species are extinct :(. The second equilibrium is where the population of the species are in perfect balance.

The other kind of object we can look for are what are called *nullclines*: locations for which either $\frac{d}{dt}v = 0$ or $\frac{d}{dt}u = 0$, but not necessarily both at the same time. This shows that the solution curve is either horizontal or vertical.

2.3.1 Variants of Lotka-Volterra

Of course, one can consider more complicated models. One clear issue with the modeling we employed in Lotka-Volterra is the fact that the Rabbit population will grow exponentially in the absence of the foxes. A reasonable fix is to use a logistic model for the Rabbit population

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} (a - \alpha u(t) - bv(t))u(t) \\ (cu(t) - d)v(t) \end{pmatrix}. \tag{2.43a}$$

$$\begin{pmatrix} u(0) \\ u(0) \end{pmatrix} = \begin{pmatrix} u_0 \\ u_0 \end{pmatrix}. \tag{2.43b}$$

There are now three equilibria to this system:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{a}{\alpha} \\ 0 \end{pmatrix}, \quad \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{d}{c} \\ \frac{a - \frac{\alpha d}{c}}{b} \end{pmatrix}. \tag{2.44}$$

Note that if $a < \frac{\alpha d}{c}$, then the third equilibrium would only appear for negative values of the fox population, which is obviously non-physical. A more detailed analysis shows that if $a < \frac{\alpha d}{c}$, the foxes are doomed to go extinct no matter what the initial condition is. This is because the Rabbits do not have enough resources to support a population large enough to support a Fox population.

There are also models involving, for example, two herbivore species in competition for the same limited resources:

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} (a - \alpha u(t) - bv(t))u(t) \\ (d - \beta v(t) - cu(t))v(t) \end{pmatrix}. \tag{2.45a}$$

$$\begin{pmatrix} u(0) \\ u(0) \end{pmatrix} = \begin{pmatrix} u_0 \\ u_0 \end{pmatrix}. \tag{2.45b}$$

2.4 Linearization method for 2x2 systems

For a general system of n variables, we consider an ODE for a vector-valued variables $\mathbf{X}(t) \in \mathbb{R}^n$ evolving by a vector-field $\mathbf{F}$

$$\frac{d}{dt}\mathbf{X}(t) = \mathbf{F}(\mathbf{X}(t)) \tag{2.46}$$

$$\mathbf{X}(0) = \mathbf{X}_0. \tag{2.47}$$

The equilibria states are those states which $\mathbf{F}(\mathbf{X}_*) = 0$. It again makes sense to use the linearization method: hence writing $\mathbf{X}(t) = \mathbf{X}_* + \varepsilon \mathbf{f}(t)$, we have

$$\frac{d}{dt}\mathbf{f}(t) = \frac{1}{\varepsilon}\mathbf{F}(\mathbf{X}_* + \varepsilon \mathbf{f}(t)) \tag{2.48}$$

$$\mathbf{f}(0) = \mathbf{f}_0. \tag{2.49}$$

To derive the linearized system, we want to send $\varepsilon \to 0$, which is the same as differentiation. Hence, (see Appendix B),

$$\frac{d}{dt}\mathbf{f}(t) = D\mathbf{F}(\mathbf{X}_*)\mathbf{f}(t) \tag{2.50}$$

$$\mathbf{f}(0) = \mathbf{f}_0. \tag{2.51}$$

where $D\mathbf{F}$ is the Jacobian matrix with entries

$$(D\mathbf{F})_{ij} = \partial_{x_j} F_i. \tag{2.52}$$

Hence, the linearization method leads us to studying linear systems.

2.4.1 2x2 linear systems

The general form of a 2×2 linear ODE is the following:

$$\frac{d}{dt}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \tag{2.53a}$$

$$\begin{pmatrix} x(0) \\ y(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{2.53b}$$

Let's first consider the case that $c = b = 0$:

$$\frac{d}{dt}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & d \end{pmatrix}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \tag{2.54}$$

$$\begin{pmatrix} x(0) \\ y(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{2.55}$$

In this case, each of the variables can be solved independently, giving

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} x_0 e^{at} \\ y_0 e^{dt} \end{pmatrix}. \tag{2.56}$$

For solving (2.53) in general, one approach is to find a change of unknowns so that we get a new system that is diagonal. See Appendix C for a refresher on diagonalization. Suppose that A is *real diagonalizable*, that is, we have real eigenvalues $\lambda_1, \lambda_2 \in \mathbb{R}$ and $P \in \mathbb{R}^{2\times 2}$ invertible such that

$$A = P\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} P^{-1}. \tag{2.57}$$

Then, applying P^{-1} to both sides of (2.53),

$$\frac{d}{dt}P^{-1}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} P^{-1}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix}. \tag{2.58}$$

Our new variables are then

$$\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = P^{-1}\begin{pmatrix} x(t) \\ y(t) \end{pmatrix}. \tag{2.59}$$

and we have

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} \tag{2.60}$$

$$\begin{pmatrix} u(0) \\ v(0) \end{pmatrix} = P^{-1}\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{2.61}$$

Solving this system gives

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = P\begin{pmatrix} e^{\lambda_1 t} & 0 \\ 0 & e^{\lambda_2 t} \end{pmatrix} P^{-1}\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{2.62}$$

The following notation is sometimes used

$$e^{At} = P\begin{pmatrix} e^{\lambda_1 t} & 0 \\ 0 & e^{\lambda_2 t} \end{pmatrix} P^{-1}. \tag{2.63}$$

Example 2 (Lotka-Volterra near extinction)**.** Let's consider the example of the Lotka-Volterra system (2.45a) linearized around $(u, v) = (0, 0)$. In this case, the linear system is

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} au(t) \\ -dv(t) \end{pmatrix}. \tag{2.64}$$

$$\begin{pmatrix} u(0) \\ u(0) \end{pmatrix} = \begin{pmatrix} u_0 \\ u_0 \end{pmatrix}. \tag{2.65}$$

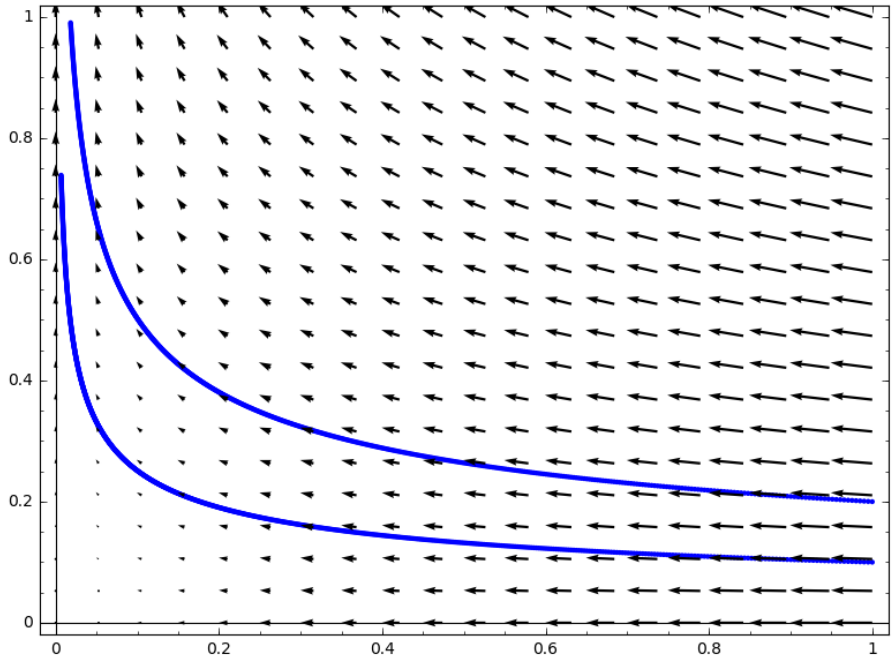

Figure 4: The linearized dynamics of Lotka-Volterra near the extinction state $(u, v) = (0, 0)$.

The system is diagonal with one positive eigenvalue ($\lambda_1 = a$) and one negative eigenvalue ($\lambda_2 = -d$). The equilibria is *unstable*, because solutions to this system diverge from the origin. However, it is more specific than this. The Rabbit population grows exponentially, whereas the population of foxes decreases exponentially, i.e. the foxes go extinct and the rabbits flourish. This is because the foxes do not have enough food to survive on so few rabbits but the rabbits are just fine. Trajectories are hyperbolas (see the lab); such an equilibrium is called a *hyperbolic point* or a *saddle point*.

As we know from linear algebra, not all real-valued matrices are real diagonalizable. Suppose that the matrix A has complex eigenvalues

$$\lambda_1 = \lambda + i\omega \tag{2.66}$$
$$\lambda_2 = \lambda - i\omega. \tag{2.67}$$

We will show the following:

Theorem 2.3. *Let A have complex eigenvalues*

$$\lambda_1 = \lambda + i\omega \tag{2.68}$$
$$\lambda_2 = \lambda - i\omega. \tag{2.69}$$

Then there is a real matrix P such that

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = P \begin{pmatrix} e^{\lambda t}\cos\omega t & e^{\lambda t}\sin\omega t \\ -e^{\lambda t}\sin\omega t & e^{\lambda t}\cos\omega t \end{pmatrix} P^{-1} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{2.70}$$

Proof. This follows from Theorem C.1. Indeed, set

$$\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = P^{-1} \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}. \tag{2.71}$$

Then,

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} \lambda & \omega \\ -\omega & \lambda \end{pmatrix}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix}. \tag{2.72}$$

One can directly check (this can be derived more systematically, but this is one of the tricks it pays to just remember in differential equations)

$$\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = e^{\lambda t}\begin{pmatrix} \cos\omega t u_0 + \sin\omega t v_0 \\ -\sin\omega t u_0 + \cos\omega t v_0 \end{pmatrix}. \tag{2.73}$$

The result follows after undoing the change of coordinates. □

Example 3 (Lotka-Volterra at balanced equilibrium)**.** Next, let us linearize (2.45a) around the equilibrium balanced equilibrium

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{d}{c} \\ \frac{a}{b} \end{pmatrix}. \tag{2.74}$$

The linearized system is

$$\frac{d}{dt}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = \begin{pmatrix} 0 & -\frac{bd}{c} \\ \frac{ca}{b} & 0 \end{pmatrix}\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} \tag{2.75}$$

The eigenvalues are

$$\lambda_{\pm} = \pm i\sqrt{ad}. \tag{2.76}$$

It follows from Theorem 2.3 that there is some invertible matrix P such that the following holds for $\omega = \sqrt{ad}$:

$$\begin{pmatrix} u(t) \\ v(t) \end{pmatrix} = P\begin{pmatrix} \cos\omega t & \sin\omega t \\ -\sin\omega t & \cos\omega t \end{pmatrix}P^{-1}. \tag{2.77}$$

It follows that the equilibrium is *stable*. However, small disturbances of the equilibrium do *not* actually converge back to equilibrium. Instead, they circulate around the equilibrium, neither getting closer nor getting further on average. Sometimes this is called *neutrally stable*.

3 General abstract theory for ODEs

Let $\mathbf{F}(t, \mathbf{x}) : [0, T] \times D \to \mathbb{R}^d$ be a known vector field and consider the ODE

$$\frac{d}{dt}\mathbf{X}(t) = \mathbf{F}(t, \mathbf{X}(t)) \tag{3.1a}$$

$$\mathbf{X}(0) = \mathbf{x}_0. \tag{3.1b}$$

Any potential solution is a curve in $\mathbb{R}^d$, $\mathbf{X}(t) : [0, T] \to \mathbb{R}^d$. In general, a solution may not exist. The theorem (which is not stated with optimal conditions) is the following. For simplicity I will just state the theorem on $\mathbb{R}^d$.

Theorem 3.1 (Cauchy-Lipschitz)**.** *Let $\mathbf{F}(t, \mathbf{x}) : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ be continuously differentiable (i.e. the vector-field $D\mathbf{F}$ exists and is continuous). Then, there is a $T_* \leq T$ such that there is a* unique *solution to the initial-value problem* (3.1)*. Moreover, if $T_* < T$ then $\lim_{t \nearrow T_*} \|\mathbf{X}(t)\| = \infty$. If $T = \infty$ and there exists a constant $C > 0$ such that $\|\mathbf{F}(t, \mathbf{x})\| \leq C$ for all $t, \mathbf{x}$, then $T_* = \infty$.*

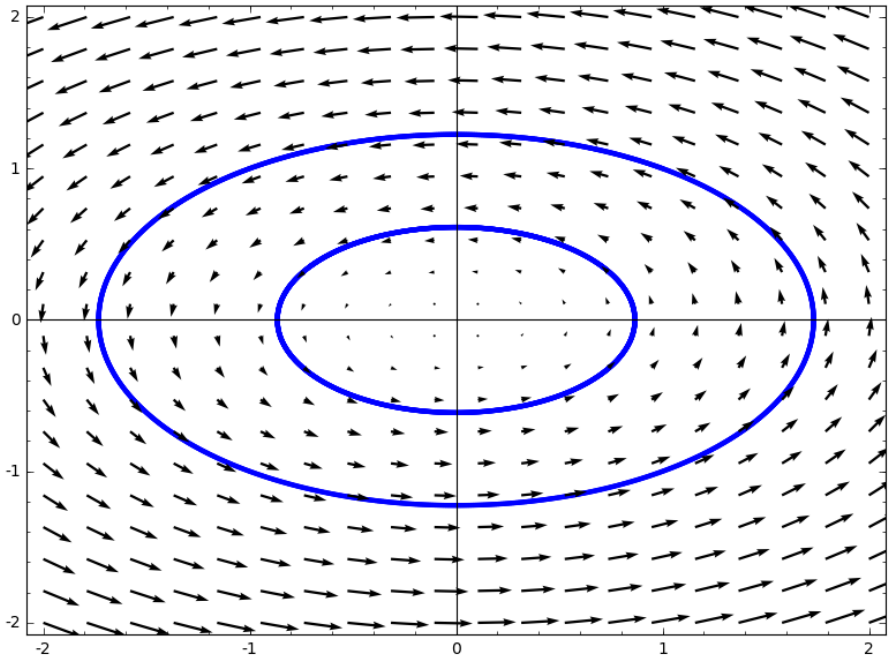

Figure 5: The dynamics of linearized Lotka-Volterra near the balanced equilibrium for some choice of parameters.

The proof of Theorem 3.1 requires a little bit of undergraduate level real analysis; see [BN12] for a proof. Notice that this theorem says that the only way the solution doesn't last for as long as possible, is if it escapes to infinity. This can happen.

Exercise 3.1. Consider the scalar ODE:

$$\frac{d}{dt}f(t) = f^2(t) \tag{3.2a}$$

$$f(0) = 1. \tag{3.2b}$$

Show that the solution to the initial value problem is $f(t) = \frac{1}{1-t}$, which as a vertical asymptote at $t = 1$, and so has no physical meaning for $t \geq 1$.

4 Numerical methods for ODEs

Let $\mathbf{F}(t, \mathbf{x}) : [0, T] \times D \to \mathbb{R}^d$ be a known vector field and consider the ODE

$$\frac{d}{dt}\mathbf{X}(t) = \mathbf{F}(t, \mathbf{X}(t)) \tag{4.1}$$

$$\mathbf{X}(0) = \mathbf{x}_0. \tag{4.2}$$

Solving equations such as this is generally impossible, even for relatively simple $\mathbf{F}$. Hence, in order to get useful answers from our mathematical model, we are going to have to consider solving this approximately with a computer. The mathematical tools for computing approximations to differential equations and similar mathematical models are known as *numerical methods* and the field is called *scientific computing* or just *numerics*. It is a wide and important field, we will only

discuss a few of the simplest numerical methods, some of them already in use for several hundred years before the invention of computers[2]..

Divide the time interval $[0, T]$ into an evenly chosen set of times $t_n = nh$, $0 \leq n \leq N$, $h = T/N$ and $t_0 = 0$, $t_1 = h$, ... $t_N = T$. We will build our numerical approximation $\{\mathbf{X}_n\}_{0 \leq n \leq N}$ with the hope that $\mathbf{X}_n \approx \mathbf{X}(t_n)$. A reasonable first approach is known as *Euler's method*, which is to approximate the time-derivative on the left-hand side of the ODE as a difference quotient:

$$\frac{\mathbf{X}_{n+1} - \mathbf{X}_n}{h} = \mathbf{F}(t_n, \mathbf{X}_n) \tag{4.3}$$

$$\mathbf{X}_0 = \mathbf{x}_0. \tag{4.4}$$

Re-arranging gives

$$\mathbf{X}_{n+1} = \mathbf{X}_n + h\mathbf{F}(t_n, \mathbf{X}_n) \tag{4.5}$$

$$\mathbf{X}_0 = \mathbf{x}_0. \tag{4.6}$$

It is the case that if one takes h smaller and smaller, the approximate solution approaches the exact solution $\|\mathbf{X}_n - \mathbf{X}(t_n)\| \to 0$. This property is called *convergence*.

Theorem 4.1 (Euler's method is convergent)**.** *Let $\mathbf{F}(t, \mathbf{x}) : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ be twice differentiable and let $\mathbf{X}_n$ be the solution given by Euler's method. Then for all $\mathbf{x}_0 \in \mathbb{R}^d$, there exists a constant C such that*

$$\max_{0 \leq n \leq N} \|\mathbf{X}_n - \mathbf{X}(t_n)\| \leq Ch. \tag{4.7}$$

The above theorem shows that one can use Euler's method to solve the initial-value problem for ODEs. Moreover, it even gives an estimate on, in general, approximately how accurate we can expect the method to be. That is, how large you have to pick N in order to be sure that the computed solution $\mathbf{X}_n$ is close to the exact solution (which in general is not known) $\mathbf{X}$. The proof of Theorem 4.1 also requires a little bit of undergraduate level real analysis, see [Ise09] (which in general is an excellent resource for mathematically rigorous analysis of numerical methods). However, we can guess the accuracy of the method only using Taylor's theorem. It will be convenient to introduce "Big-Oh" notation:

Definition 1 (Big-Oh)**.** Given two functions f, g we say $f = O(g)$ (pronounced f is Big-Oh of g) if there exists a constant such that

$$f(x) \leq Cg(x). \tag{4.8}$$

We will write $f_1 = f_2 + O(g)$ (and similar expressions) if $|f_1 - f_2| = O(g)$.

Let us estimate the error. the true solution $\mathbf{X}(t)$ satisfies

$$\mathbf{X}(t_{n+1}) = \mathbf{X}(t_n) + \int_{t_n}^{t_{n+1}} \mathbf{F}(s, \mathbf{X}(s))ds. \tag{4.9}$$

[2]Before computers, people had to approximately calculate solutions by hand. This was a tedious and time-consuming task but sometimes there was no other option. In the 1940s and 1950s or so, entire teams of people would be devoted to implementing numerical methods by hand in order to calculate the results required to do all sorts of things, for example, flight trajectories of space craft.

Hence,

$$\mathbf{X}_{n+1} - \mathbf{X}(t_{n+1}) = \mathbf{X}_n - \mathbf{X}(t_n) + h\mathbf{F}(t_n, \mathbf{X}_n) - \int_{t_n}^{t_{n+1}} \mathbf{F}(s, \mathbf{X}(s))ds. \tag{4.10}$$

Then, we use Taylor's theorem to show that

$$\mathbf{F}(s, \mathbf{X}(s))ds = \mathbf{F}(t_n, \mathbf{X}(t_n)) + (s - t_n)\partial_t \mathbf{F}(t_n, \mathbf{X}(t_n)) + (s - t_n)D\mathbf{F}(t_n, \mathbf{X}(t_n))\frac{d}{dt}\mathbf{X}(t_n) + O\left(|s - t_n|^2\right). \tag{4.11}$$

Hence,

$$\mathbf{X}_{n+1} - \mathbf{X}(t_{n+1}) = \mathbf{X}_n - \mathbf{X}(t_n) + h\left(\mathbf{F}(t_n, \mathbf{X}_n) - \mathbf{F}(t_n, \mathbf{X}(t_n))\right) + O(h^2). \tag{4.12}$$

Then, because $\mathbf{F}$ is continuously differentiable,

$$\mathbf{F}(t_n, \mathbf{X}_n) - \mathbf{F}(t_n, \mathbf{X}(t_n) = D\mathbf{F}(t_n, \mathbf{X}_n)(\mathbf{X}_n - \mathbf{X}(t_n)) + O(\|\mathbf{X}_n - \mathbf{X}(t_n)\|^2). \tag{4.13}$$

Therefore, we can deduce that there exists a constants $K_1, K_2 > 0$ such that

$$\|\mathbf{X}_{n+1} - \mathbf{X}(t_{n+1})\| \leq (1 + K_1 h)\|\mathbf{X}_n - \mathbf{X}(t_n)\| + K_2 h^2. \tag{4.14}$$

This is an estimate on how rapidly the error between the exact solution and our numerical calculation grows. Notice that this has nothing to do with the round-off error of the computer, i.e. the fact that computers do not do exact arithmetic. This error would be present even with perfect arithmetic. We see several things:

1. Each time-step loses at least $K_2 h^2$ accuracy. Since there are $N = T/h$ time steps, this suggests an error at best like $O(h)$.

2. The error in general is amplified exponentially a tiny bit every time-step. For many systems this is actually not true, and the estimate we have obtained is very sub-optimal. However, for some systems, there is exponential growth of error. For some systems (for example, the weather) we will never be able to accurately compute the solution very far in advance.

3. The error estimate (4.14) is the first step in the proof of Theorem 4.1.

4.1 Stiffness and implicit methods

There was something arbitrary in how we designed the Euler method. After all, we could have chosen

$$\frac{\mathbf{X}_{n+1} - \mathbf{X}_n}{h} = \mathbf{F}(t_{n+1}, \mathbf{X}_{n+1}) \tag{4.15}$$

$$\mathbf{X}_0 = \mathbf{x}_0. \tag{4.16}$$

This method is called *backward Euler*. One can show that Theorem 4.1 also applies to backward Euler as well. It is not any more accurate than (forward) Euler. However, it is in general much more difficult to code, because it is not usually possible to easily solve for $\mathbf{X}_{n+1}$. Methods of this type are called *implicit*. Methods of the forward Euler type are called *explicit*. Usually one needs to numerically solve for $\mathbf{X}_{n+1}$ using a numerical linear or nonlinear system solver (depending on

whether $\mathbf{F}$ is linear in $\mathbf{X}_{n+1}$ or not). Given that it is not any more accurate, but is way harder to implement, there must be another reason I brought up this method.

Consider the very simple scalar linear equation:

$$\frac{d}{dt}y(t) = -100y(t) \tag{4.17}$$

$$y(0) = 1. \tag{4.18}$$

The exact solution is $y(t) = e^{-100t}$. Consider now solving this with Euler's method:

$$y_{n+1} = y_n - 100hy_n = (1 - 100h)y_n, \tag{4.19}$$

and hence Euler's method gives the solution

$$y_n = (1 - 100h)^n. \tag{4.20}$$

Hence, we see a fundamental problem: if $1 - 100h < -1$, that is, if $h > 1/50$, then the numerical solution is actually *growing* (and oscillating!). This looks absolutely nothing like the rather boring exact solution, which is, for intents and purposes, essentially zero for $t > 1/50$. This is an example of what is called a *numerical instability*: an "instability" observed numerically which is not actually there physically. Notice that there is nothing wrong with Euler's method per se: the method really is convergent, so when you make h small enough, the issue goes away. The problem is appearing for h which is small, but not quite small enough to "resolve" the real dynamics. This problem is called *stiffness*, and problems which have this kind of character are called *stiff* (see [Ise09] for a more detailed study). Stiffness is the fundamental reason why anyone cares about implicit methods. Indeed, consider now backward Euler:

$$y_{n+1} = y_n - 100hy_{n+1}, \tag{4.21}$$

and hence

$$y_{n+1} = \frac{y_n}{1 + 100h}. \tag{4.22}$$

This solution decays exponentially *regardless* of h and so it does not suffer from the same problem as forward Euler. As a general rule, implicit methods handle stiffness better than explicit methods. In fact, no explicit method can handle stiffness very well. A problem may be so stiff that it is actually more efficient to use the slow and difficult to implement implicit methods, which allow a larger time-step, than it is to use a simpler explicit methods (which would require prohibitively small time-steps).

4.2 Higher order methods

One can design methods which are more accurate than $O(h)$. The first such an example is the *implicit trapezoidal method*

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \frac{h}{2}\mathbf{F}(t_{n+1}, \mathbf{X}_{n+1}) + \frac{h}{2}\mathbf{F}(t_n, \mathbf{X}_n). \tag{4.23}$$

One can show that this method is $O(h^2)$ accurate.

Definition 2. We say that a method is p-th order accurate if Theorem 4.1 holds for the method with an error estimate of the form

$$\max_{0 \le n \le N} \|\mathbf{X}_n - \mathbf{X}(t_n)\| \le Ch^p. \tag{4.24}$$

Hence, implicit trapezoidal is second-order accurate. That extra accuracy can go a long way. There is also an explicit analogue of the trapezoidal method, called *explicit trapezoidal method*

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \frac{h}{2}\mathbf{F}\left(t_{n+1}, \mathbf{X}_n + h\mathbf{F}(t_n, \mathbf{X}_n)\right) + \frac{h}{2}\mathbf{F}(t_n, \mathbf{X}_n). \tag{4.25}$$

This is a very similar method but one which we have approximated the implicit part by an explicit time step. As it turns out, this method is still $O(h^2)$ accurate, despite this approximation. As with the Euler method, the implicit trapezoidal method performs better on "stiff" problems than explicit trapezoidal.

There are a number of higher order accurate schemes. The matlab routine ode45 is a combination of 4-th order and 5-th order explicit methods. The answers are compared to one another to adjust h, the step-size, on the fly in order to automatically estimate whether or not h is small enough to get an accurate answer – adaptive refinement is very useful and is a much more advanced topic in numerical algorithms. The matlab routine ode23 is a similar combination of second and third order implicit methods. The difference in accuracy is due to the fact that the computational cost of implicit methods goes up rapidly as you go to higher order. For most applications the third order accuracy of ode23 is good enough to get by and is still fast enough to execute in a reasonable amount of time. If your problem does not have stiffness problems, you should use ode45 and if there are stiffness problems you should use ode23.

5 Diffusion: introduction to PDE

Suppose now we are interested in a density of chemical or a population of a species as a function of both space and time, denoted $\rho(t, x)$. Let us consider the case $x \in \mathbb{R}$ for now. Now, we are interested in understanding the total amount of material in the the interval $[a, b]$:

$$\frac{d}{dt}\int_a^b \rho(t, x)dx = \int_a^b \partial_t \rho(t, x)dx. \tag{5.1}$$

Let us suppose for now that the total amount of material is conserved, and so material can only enter and exit at the boundary of the interval. Hence, we can suppose there is a function $F(t, x)$, called the *flux*, which specifies flow into and out of the interval. The flux tells how material is moving in space and time, different mathematical models for the physics/biology/chemistry give different F. By convention, the sign is take so that

$$\frac{d}{dt}\int_a^b \rho(t, x)dx = -F(t, b) + F(t, a) = -\int_a^b \partial_x F(t, x)dx. \tag{5.2}$$

The second equality was the fundamental theorem of calculus. Since this is supposed to hold for all $[a, b]$ we get that in fact

$$\partial_t \rho(t, x) + \partial_x F(t, x) = 0. \tag{5.3}$$

The first model we will consider is the case that density tends to move from high concentration to low concentration. The simplest flux that does this is just that the flux is inversely proportional to the slope: for some constant $\kappa > 0$,

$$F(t, x) = -\kappa \partial_x \rho(t, x). \tag{5.4}$$

Here $\kappa > 0$ is a positive constant called the *diffusivity*, and it depends on the exact physics/biology/chemistry. This leads to our first *partial differential equation*, a PDE known as the *diffusion equation*

$$\partial_t \rho = \kappa \partial_{xx} \rho \tag{5.5}$$

$$\rho(0, x) = \rho_0(x). \tag{5.6}$$

As before, we need also an initial distribution. A thorough study of this elegant PDE would require more time and mathematical background than we have, but we can learn a little about the equation just by observing two of its exact solutions and also through numerical studies in the lab.

Example 4. The following function

$$\rho(t, x) = \frac{1}{(4\pi\kappa t)^{1/2}} e^{-\frac{|x|^2}{4\kappa t}}. \tag{5.7}$$

solves $\partial_t \rho = \kappa \partial_{xx} \rho$ (the initial data is a little hard to make sense of). This function is concentrated on an interval of width $\approx \sqrt{\kappa t}$ with height $\approx (\kappa t)^{-1/2}$ (note that $\int_{-\infty}^{\infty} \rho(t, x) dx = 1$). Hence, the solution is concentrated for short times and becomes increasingly more and more spread out (hence the term "diffusion").

Example 5. The following function solves

$$\rho(t, x) = e^{-\kappa 4\pi^2 n^2 t} \sin 2\pi n x. \tag{5.8}$$

solves $\partial_t \rho = \kappa \partial_{xx} \rho$. This function keeps its shape of a sin curve of frequency $2\pi n$ but decays exponentially. It is interesting to note that the decay is faster for n larger (and this rate of increase is quite rapid). The diffusion equation tends to wipe out rapid oscillations whereas larger ripples take longer to decay.

The diffusion equation is a universal model in that it is a decent model for a wide range of physical, biological, and chemical phenomena. It accurately models heat diffusing through a homogeneous medium. It accurately models a chemical solvent as it diffuses through a liquid at rest. It accurately models a population of organisms that are executing a "random walk" that is, the velocity at any given time can be considered a random variable that does not depend on the past choices and chosen with a bell-curve distribution (this turns out to be a reasonable approximation for some organisms, but not others).

For bacteria, the diffusion equation is actually a pretty good approximation. If we then add in logistic population dynamics, we get a type of equation called a *reaction-diffusion equation*

$$\partial_t \rho = \kappa \partial_{xx} \rho + \lambda (1 - \frac{1}{A} \rho) \rho \tag{5.9}$$

$$\rho(0, x) = \rho_0(x). \tag{5.10}$$

This population will diffuse while at each point, the population will try to tend to $\rho = A$. This is a reasonable first model for the expansion of some bacteria populations in a petri dish. The bacteria will randomly explore while dividing and using the resources. This model does not have a boundary in it yet, so the bacteria population will continue expanding forever (but at each location at space, the population will settle to $\rho(t, x) = A$. If we want to study the model on an interval $[a, b]$ then we need a boundary condition. If we want to impose that the population cannot enter or exit from the boundary (for example, the side of a petri dish), then we need to ensure that there is no flux into or out of the boundary, i.e.

$$\partial_x \rho(t, a) = \partial_x \rho(t, b) = 0. \tag{5.11}$$

The equations then become

$$\partial_t \rho = \kappa \partial_{xx} \rho + \lambda(1 - \frac{1}{A}\rho)\rho \quad x \in (a, b) \tag{5.12}$$

$$\rho(0, x) = \rho_0(x) \quad x \in (a, b) \tag{5.13}$$

$$\partial_x \rho(t, a) = \partial_x \rho(t, b) = 0. \tag{5.14}$$

Multiple interacting populations is also possible. For example, the following variant of Lotka-Volterra:

$$\partial_t u = \kappa_1 \partial_{xx} u + (a - bv)u \tag{5.15a}$$

$$\partial_t v = \kappa_2 \partial_{xx} v + (cu - d)v \tag{5.15b}$$

$$u(0, x) = u_0(x) \tag{5.15c}$$

$$v(0, x) = v_0(x). \tag{5.15d}$$

This would model the tendency for foxes and rabbits to also tend to disperse from high population densities as they hunted or foraged (respectively).

5.1 Linearization method for PDE

The Lotka-Volterra-diffusion system (5.15) still has the balanced equilibrium solution $u(t, x) = \frac{d}{c}$, $v(t, x) = \frac{a}{b}$. Let us investigate the stability of the balanced equilibrium, which gives linearized system

$$\partial_t u = \kappa_1 \partial_{xx} u - \frac{d}{c} v \tag{5.16}$$

$$\partial_t v = \kappa_2 \partial_{xx} v + \frac{a}{b} u. \tag{5.17}$$

This is still a PDE so we cannot immediately use the ODE linearization method. However, we can if we just ask about solutions of the specific form

$$\begin{pmatrix} u(t, x) \\ v(t, x) \end{pmatrix} = e^{\lambda t} \begin{pmatrix} u_0 \sin nx \\ v_0 B \sin nx \end{pmatrix}, \tag{5.18}$$

then we can reduce the PDE stability problem down to an ODE stability problem! Similar arguments apply also for $\cos nx$. Plugging this into the PDE gives

$$\lambda \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} -\kappa_1 n^2 & -\frac{d}{c} \\ \frac{a}{b} & -\kappa_2 n^2 \end{pmatrix} \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}. \tag{5.19}$$

This is an eigenvalue problem! Hence, we 're looking for λ such that

$$0 = (-\lambda - \kappa_1 n^2)(-\lambda - \kappa_2 n^2) + \frac{ad}{cb} \tag{5.20}$$

$$0 = \lambda^2 + \lambda n^2(\kappa_1 + \kappa_2) + \kappa_1 \kappa_2 n^4 + \frac{ad}{cb}. \tag{5.21}$$

The eigenvalues of this are given by

$$2\lambda_\pm = -n^2(\kappa_1 + \kappa_2) \pm \sqrt{n^4(\kappa_1 + \kappa_2)^2 - 4\left(\kappa_1 \kappa_2 n^4 + \frac{ad}{cb}\right)}. \tag{5.22}$$

These eigenvalues both have a negative real part, which means that solutions will spiral back to the balanced equilibrium. It is a deep fact that actually this calculation implies stability for *all* perturbations, not just those of sin and cosine type. Hence, the addition of spatial diffusion changed the dynamics: we now see damped oscillations back to balanced equilibrium whereas before we saw pure oscillations. This is due to the tendency for diffusion to eliminate spatial oscillations.

6 Numerical methods for diffusion and reaction-diffusion

Let us consider trying to solve the diffusion equation numerically:

$$\partial_t \rho = \kappa \partial_{xx} \rho \qquad x \in [0,1] \tag{6.1}$$
$$\rho(0,x) = \rho_0(x) \tag{6.2}$$
$$\partial_x \rho(t,0) = \partial_x \rho(t,1) = 0. \tag{6.3}$$

I have changed the problem a little. I have decided to study over one interval and I imposed *zero-flux boundary conditions* at $x = 0, 1$. These conditions ensure that density cannot enter or leave the interval. As in the case of ODEs divide the time interval $[0,T]$ into an evenly chosen set of times $t_n = nh$, $0 \leq n \leq N_h$, $h = T/N$ and $t_0 = 0$, $t_1 = h$, ... $t_N = T$. Now we also have to divide the space interval $[0,1]$ into an evenly chosen set of locations $x_n = nk$, $0 \leq n \leq N_k$, $k = 1/N_k$ and $x_0 = 0$, $x_1 = h$, ... $x_{N_k} = 1$. Our numerical solution will now be a double indexed set $\rho_{n,j} \approx \rho(nh, jk) = \rho(t_n, x_j)$. For any three-times differentiable function, Taylor's theorem gives (a good exercise!)

$$\partial_{xx} f(x) = \frac{f(x+k) - 2f(x) + f(x-k)}{k^2} + O(k^2). \tag{6.4}$$

Hence, it makes sense to replace the diffusion equation with the PDE equivalent of forward Euler:

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{h} = \frac{\kappa}{k^2} \left(\rho_{n,j+1} - 2\rho_{n,j} + \rho_{n,j+1} \right) \quad 0 \leq j \leq N_k. \tag{6.5}$$
$$\rho(0,x) = \rho_0(x) \tag{6.6}$$
$$\rho_{n,-1} = -\rho_{n,1} \tag{6.7}$$
$$\rho_{n,N_k+1} = -\rho_{n,N_k-1}. \tag{6.8}$$

Note the latter two strange conditions are approximations to the boundary conditions. They are used to resolve the problems at the boundaries in the first equation at $j = 0, N_k$.

6.1 Numerical stability for PDE methods

Let us ignore boundary conditions for a moment and consider just the pure numerical method

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{h} = \frac{\kappa}{k^2} \left(\rho_{n,j+1} - 2\rho_{n,j} + \rho_{n,j+1} \right). \tag{6.9}$$

The method looks like it should work – but is there a stiffness issue? There is a convenient way of investigating this, known as von Neumann analysis (at least if you ignore boundary conditions). Let us look for solutions of the form (a reasonable thing to do if we remember the exact solutions above!):

$$\rho_{n,j} = \lambda^n \sin \omega j. \tag{6.10}$$

Then note that by trig identities,

$$\sin\omega(j+1) = \sin\omega\cos\omega j + \cos\omega\sin\omega j \tag{6.11}$$

$$\sin\omega(j-1) = -\sin\omega\cos\omega j + \cos\omega\sin\omega j. \tag{6.12}$$

Hence,

$$\lambda - 1 = \frac{\kappa h}{k^2}(2\cos\omega - 2) \tag{6.13}$$

Now you might point out that not every value of ω really makes sense – the grid is only so fine, so the fastest kind of oscillation we can really represent is one which oscillates like $+1$, -1, $+1$, -1. This suggests an $\omega = \pi$ (or more accurately, we can choose $\omega \in [-\pi, \pi]$). If we use $\omega = \pi$, then

$$\lambda = 1 - 4\frac{\kappa h}{k^2}. \tag{6.14}$$

The numerical solution is going to decay if

$$-1 < 1 - 4\frac{\kappa h}{k^2} < 1,$$

and hence we get the restriction

$$\frac{\kappa h}{k^2} < \frac{1}{2}. \tag{6.15}$$

This is sometimes called *a CFL condition* (for Courant-Friedrichs-Lewy). We can view this as a time-step restriction on h in terms of κ and k. One can prove the following theorem (see [Str04] for the proof, and a general introduction to numerical methods PDEs).

Theorem 6.1. *If* (6.15) *is satisfied, then the forward Euler-central difference numerical approximation $\rho_{n,j}$ satisfies*

$$\max_{0\le n\le N_h}\max_{0\le j\le N_k}|\rho_{n,j} - \rho(t_n, x_j)| = O(h) + O(k^2). \tag{6.16}$$

In order to obtain both better accuracy and better stability, we can use the implicit trapezoidal (which for some reason in this context is called Crank-Nicolson (CR)):

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{h} = \frac{\kappa}{2k^2}\left(\rho_{n+1,j+1} - 2\rho_{n+1,j} + \rho_{n+1,j+1}\right) + \frac{\kappa}{2k^2}\left(\rho_{n,j+1} - 2\rho_{n,j} + \rho_{n,j+1}\right). \tag{6.17}$$

This method does not have a time-step restriction.

Theorem 6.2. *The Crank-Nicolson scheme numerical approximation $\rho_{n,j}$ satisfies*

$$\max_{0\le n\le N_h}\max_{0\le j\le N_k}|\rho_{n,j} - \rho(t_n, x_j)| = O(h^2) + O(k^2). \tag{6.18}$$

Note that CR requires solving a linear system each time-step. However, solving linear systems is much easier than solving nonlinear ones, and moreover, this particular kind of linear system can be solved especially efficiently. For this reason it is often much faster to use implicit methods such as this for diffusion problems.

In order to extend the methods to reaction diffusion equations, consider a problem of the form

$$\partial_t\rho = \kappa\partial_{xx}\rho + F(\rho). \tag{6.19}$$

As it happens, it is usually the case that the reaction term F does not contain any stiffness problems. One method that is very reliable is a mixed backward-forward Euler:

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{h} = \frac{\kappa}{k^2}\left(\rho_{n+1,j+1} - 2\rho_{n+1,j} + \rho_{n+1,j+1}\right) + F(\rho_{n,j}). \tag{6.20}$$

Designing higher order accurate methods that both (A) have no time-step restriction and (B) don't require solving a nonlinear system is a little more complicated.

7 Transport in fluid mechanics

7.1 Scalar transport in 1D

In this section we are interested in understanding another fundamental physical phenomenon that requires a PDE to model: *transport*. Let us start as in Section 5 and consider a density $\rho(t,x)$ a function of time and space. Suppose that we are not yet modeling the creation or destruction of the density, and so for some flux function $F(t,x)$ we have, as in (5.2):

$$\frac{d}{dt}\int_a^b \rho(t,x)dx = -F(t,b) + F(t,a) = -\int_a^b \partial_x F(t,x)dx. \tag{7.1}$$

Now, we suppose that, instead of diffusing, the material is just being carried along by a known velocity field $v(t,x)$; convention has it that $v > 0$ means the density is being transported to the right, whereas $v < 0$ means the density is being transported to the left. Therefore, the flux at a point is the rate at which material is being transported from the right to the left:

$$F(t,x) = v(t,x)\rho(t,x).$$

This leads to the PDE known as the *transport equation*

$$\partial_t \rho + \partial_x(v\rho) = 0 \tag{7.2}$$
$$\rho(0,x) = \rho_0(x). \tag{7.3}$$

We are assuming right now that we already know the velocity field v, and we only need to solve for ρ.

A mathematical study of this PDE is also pretty complicated, but in some ways a little less complicated than the diffusion equation. One case we can begin with is when the velocity is just a constant: $v(t,x) = v \in \mathbb{R}$, i.e. the density is being transported at a constant velocity v. Here, the equation is

$$\partial_t \rho + v\partial_x \rho = 0 \tag{7.4a}$$
$$\rho(0,x) = \rho_0(x). \tag{7.4b}$$

If we think about a single particle $X(t,\alpha)$ moving with constant velocity v starting at location α, i.e.

$$\frac{d}{dt}X(t,\alpha) = v \tag{7.5}$$
$$X(0,\alpha) = \alpha, \tag{7.6}$$

then we have

$$X(t,\alpha) = \alpha + tv. \tag{7.7}$$

This physical system describes the motion of a single particle in a constant velocity. If we imagine the density is just being transported, then if you follow this same path, maybe we expect the density to be constant. This is precisely right! By the chain rule:

$$\frac{d}{dt}\left(\rho(t,\alpha+tv)\right) = \partial_t\rho(t,\alpha+tv) + v\partial_x\rho(t,\alpha+tv) = 0. \tag{7.8}$$

Therefore, the density at the location $\alpha + tv$ is the same as at $t = 0$,

$$\rho(t, \alpha + tv) = \rho_0(\alpha). \tag{7.9}$$

If you want to know $\rho(t, x)$ as a function of x, you then finally have to solve $x = \alpha + tv$ for α. This gives

$$\rho(t, x) = \rho_0(x - tv). \tag{7.10}$$

This is the general solution of (7.4) – we have succeeded in solving our first PDE.

7.1.1 The method of characteristics

We have also learned a hint about how to solve transport equations in general – the method, which is actually a more general method that can solve a whole class of different PDEs, is called *the method of characteristics*. Consider now the general scalar case:

$$\partial_t \rho(t, x) + v(t, x)\partial_x \rho(t, x) = -\partial_x v(t, x)\rho(t, x) \tag{7.11a}$$
$$\rho(0, x) = \rho_0(x). \tag{7.11b}$$

The particles are solving

$$\frac{d}{dt} X(t, \alpha) = v(t, X(t, \alpha)) \tag{7.12}$$
$$X(0, \alpha) = \alpha. \tag{7.13}$$

The solution can always be found (assuming the velocity field v isn't messed up), but not by hand, you'd need a computer. We then have

$$\frac{d}{dt} \left(\rho(t, X(t, \alpha))\right) = -\partial_x v(t, X(t, \alpha))\rho(t, X(t, \alpha)). \tag{7.14}$$

Therefore, by integrating factors

$$\rho(t, X(t, \alpha)) = \rho_0(\alpha) \exp\left(-\int_0^t \partial_x v(s, X(s, \alpha))ds\right). \tag{7.15}$$

Then, we still have to solve $X(t, \alpha) = x$ for α. This can always been done, theoretically speaking, but you'd probably need a computer. Then the general solution is

$$\rho(t, x) = \rho_0(\alpha(t, x)) \exp\left(-\int_0^t \partial_x v(s, X(s, \alpha(t, x)))ds\right). \tag{7.16}$$

This formula, while true, is also pretty useless, because in most cases, it is way too hard to solve. There are only a few cases which can be solved. Let's just do one example:

$$v(t, x) = -x. \tag{7.17}$$

In this case, the particle trajectory is

$$\frac{d}{dt} X(t, \alpha) = -X(t, \alpha) \tag{7.18}$$
$$X(0, \alpha) = \alpha, \tag{7.19}$$

which is solved as

$$X(t,\alpha) = \alpha e^{-t}. \tag{7.20}$$

Along these trajectories

$$\frac{d}{dt}\left(\rho(t, X(t,\alpha))\right) = \rho(t, X(t,\alpha)) \tag{7.21}$$

$$\rho(0, X(0,\alpha)) = \rho_0(\alpha), \tag{7.22}$$

which solves to

$$\rho(t, X(t,\alpha)) = \rho_0(\alpha)e^t. \tag{7.23}$$

Then we can solve $x = \alpha e^{-t}$ for α and we get $\alpha = xe^t$. This gives the formula

$$\rho(t,x) = e^t \rho_0(xe^t). \tag{7.24}$$

7.2 Transport in higher dimensions

Most environments we are concerned with in the real world need to be modeled as two or three dimensional. In this section we introduce some of the basic mathematical vocabulary for doing so.

A fluid here will be represented as a (generally time-dependent) velocity field $\mathbf{u}(t,\mathbf{x})$, which we suppose satisfies $\max \|D\mathbf{u}(t,\mathbf{x})\| < \infty$. Let $\rho(t,\mathbf{x})$ be a scalar quantity, e.g. the density of a pollutant or chemical in the fluid, the population density of a microbe species swimming in the fluid, or the temperature of the fluid itself. Let $V \subset \mathbb{R}^d$ be any volume. If nothing is creating or destroying concentration then the amount of ρ in a given domain can only change by transport across the boundary of the domain . Hence, as in the 1d case, there should be a flux $\mathbf{F}(t,\mathbf{x})$ such that

$$\frac{d}{dt}\int_V \rho(t,\mathbf{x})d\mathbf{x} = -\int_{\partial V} \mathbf{F}(t,\mathbf{x})\cdot\mathbf{n}dS(n), \tag{7.25}$$

where $\mathbf{n}$ is the outward unit normal to D (the $-$ sign is convention). The flux tells us how the material is being transported in the fluid, and different mathematical models for the physics lead to different choices of $\mathbf{F}$. By the divergence theorem (B.26),

$$\frac{d}{dt}\int_V \rho(t,\mathbf{x})d\mathbf{x} = -\int_V \nabla\cdot\mathbf{F}(t,\mathbf{x})d\mathbf{x}. \tag{7.26}$$

Since this has to hold for all V, we obtain

$$\partial_t\rho + \nabla\cdot\mathbf{F} = 0. \tag{7.27}$$

To model transport, the flux takes the form

$$\mathbf{F}(t,\mathbf{x}) = \rho(t,\mathbf{x})\mathbf{u}(t,\mathbf{x}), \tag{7.28}$$

which leads to the transport equation in $\mathbb{R}^d$:

$$\partial_t\rho + \nabla\cdot(\rho\mathbf{u}) = 0. \tag{7.29}$$

In many (but not all) applications, it is physically reasonable to assume the velocity satisfies the following (called *incompressibility*):

$$\nabla \cdot \mathbf{u} = 0. \tag{7.30}$$

Under this assumption, the transport equation becomes

$$\partial_t \rho + \mathbf{u} \cdot \nabla \rho = 0. \tag{7.31}$$

Naturally, in order to compute ρ, we also need an initial condition:

$$\rho(0, \mathbf{x}) = \rho_0(\mathbf{x}). \tag{7.32}$$

If the domain in which the fluid is moving, $D \subset \mathbb{R}^d$, has boundaries, then we also might need boundary conditions. For simplicity, we will assume that (denoting ∂D the boundary of D and n the outward unit normal),

$$\mathbf{u} \cdot n|_{\partial D} = 0. \tag{7.33}$$

This ensures that the flux (7.28) satisfies $\mathbf{F} \cdot n|_{\partial D} = 0$ which means that no concentration ρ is exiting or entering the domain. This also ensures that we do not need to add any additional boundary conditions on ρ.

8 Numerical methods for transport equations

Consider first trying to use a computer to numerically approximate solutions to

$$\partial_t \rho + a \partial_x \rho = 0. \tag{8.1}$$

Following what we did in the case of the diffusion equation, it makes sense to try to use

$$\partial_t \rho_{n,j} \approx \frac{\rho_{n+1,j} - \rho_{n,j}}{h} \tag{8.2}$$

$$\partial_x \rho_{n,j} \approx \frac{\rho_{n,j+1} - \rho_{n,j-1}}{2k}. \tag{8.3}$$

Resulting in the simple explicit scheme

$$\rho_{n+1,j} = \rho_{n,j} - a\frac{h}{2k}\left(\rho_{n,j+1} - \rho_{n,j-1}\right). \tag{8.4}$$

Unfortunately, this scheme does not actually work! Indeed, let us follow the argument in Section 6.1 with a slight twist and look for a solution of the form

$$\rho_{n,j} = \lambda^n e^{i\omega j} \tag{8.5}$$

It is a little strange looking for complex-valued solutions, but it results in a much cleaner understanding. Plugging this in,

$$\lambda = 1 - \frac{ah}{2k}\left(e^{i\omega} - e^{-\omega}\right) = 1 - i\frac{ah}{k}\sin\omega. \tag{8.6}$$

The complex modulus of is then

$$|\lambda| = 1 + \frac{a^2 h^2}{k^2} \sin^2 \omega. \tag{8.7}$$

For most values of ω this is strictly larger than one and hence solutions will grow exponentially, i.e. a non-physical instability.

In order to obtain a numerical method that is stable, we have to devise a numerical method that respects the PDE more. A transport equation such as this only sends information in one direction: to the right if $a > 0$ and to the left if $a < 0$. It makes sense to devise a numerical method that respects this fact. A scheme of this type is called an *upwinding scheme*. Let as assume $a > 0$. Then the scheme we should try to use is

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{k} = -a \frac{\rho_{n,j} - \rho_{n,j-1}}{h}. \tag{8.8}$$

Giving the explicit "upwinding" scheme

$$\rho_{n+1,j} = \rho_{n,j} - a \frac{h}{k} \left(\rho_{n,j} - \rho_{n,j-1} \right). \tag{8.9}$$

Then,

$$\lambda = \left(1 - a\frac{h}{k}\right) + a\frac{h}{k} e^{-i\omega} \tag{8.10}$$

$$= 1 - a\frac{h}{k}(1 - \cos\omega) + ia\frac{h}{k}\sin\omega. \tag{8.11}$$

Then, (note the argument is using $a > 0$),

$$|\lambda|^2 = (1 - a\frac{h}{k})^2 + a\frac{h}{k}(1 - a\frac{h}{k})\cos\omega + a^2\frac{h^2}{k^2} \tag{8.12}$$

$$\leq (1 - a\frac{h}{k})^2 + a\frac{h}{k} \tag{8.13}$$

$$\leq 1 - a\frac{h}{k} + a^2\frac{h^2}{k^2}. \tag{8.14}$$

We can view this as a polynomial in $x = a\frac{h}{k}$ and we get $x^2 - x + 1 \leq 1$ precisely when $0 \leq x \leq 0$. That is, the requirement

$$a\frac{h}{k} \leq 1. \tag{8.15}$$

This is the CFL requirement for the upwinding scheme. It has a physical meaning in this context: a particle moving in the velocity field cannot travel more than one grid cell in a single time step! If $a < 0$, then we hae to change the scheme:

$$\frac{\rho_{n+1,j} - \rho_{n,j}}{k} = -a \frac{\rho_{n,j+1} - \rho_{n,j}}{h}. \tag{8.16}$$

If a depends on space and time, then the direction of the finite difference must be adapted on the fly to respect the direction that transport is occurring.

Upwinding is a convenient and physically motivated way to obtain very fast explicit schemes (it also works in higher dimensions). Obtaining robust higher order upwinding schemes is not so easy, but can be done.

9 Advection-diffusion and reaction-advection-diffusion

9.1 1D advection-diffusion and advection-diffusion reaction

By adjusting the 1D flux to contain both advection and diffusion,

$$F(t,x) = -\kappa\partial_x\rho + \rho v, \tag{9.1}$$

we get the advection-diffusion equation

$$\partial_t\rho + \partial_x(v\rho) = \kappa\partial_{xx}\rho. \tag{9.2}$$

Similarly, we can include population dynamics and chemical reactions, such as the Lotka-Volterra system:

$$\partial_t\rho + \partial_x(v\rho) = \kappa_1\partial_{xx}\rho + (a - bq)\rho \tag{9.3a}$$
$$\partial_t q + \partial_x(vq) = \kappa_2\partial_{xx}q + (c\rho - dq)q \tag{9.3b}$$
$$\rho(0,x) = \rho_0(x) \tag{9.3c}$$
$$q(0,x) = q_0(x). \tag{9.3d}$$

9.2 Advection-Diffusion and Reaction-Advection-Diffusion in higher dimensions

In higher dimensions, flux is updated to be

$$\mathbf{F}(t,\mathbf{x}) = \rho(t,\mathbf{x})\mathbf{u}(t,\mathbf{x}) - \kappa\nabla\rho(t,\mathbf{x}),$$

where $\kappa > 0$ is a positive constant that depends on the exact physics being modeled. The extra term $-\kappa\nabla\rho$ is called the diffusive flux. This Flux Law says that, if the fluid is at rest, then the concentration of ρ moves from high concentration to low concentration at a rate proportional to the concentration gradient. It gives the PDE

$$\partial_t\rho + \nabla\cdot(\rho\mathbf{u}) = \kappa\Delta\rho, \tag{9.4}$$

which is usually called the *advection-diffusion equation*. We will always assume that $\mathbf{u}\cdot n|_{\partial D} = 0$ along the boundary. Now however, due to the diffusion term $\kappa\Delta\rho$, this will not imply that the flux is vanishing at the boundary. Hence, if we want to keep the density inside the domain, we have to impose the additional boundary condition

$$\nabla\rho\cdot n|_{\partial D} = 0, \tag{9.5}$$

where n is the outward unit normal of the boundary ∂D.

9.3 Reaction-advection-diffusion

Many industrial and scientific applications of the above ideas involve one or more concentrations of interacting species of chemicals in a fluid flow. The flow is sometimes occurring in nature (for example, in the atmosphere or ocean) or the flow is engineered in a machine to, for example, attempt to maximize the rate at which chemicals react. For example, suppose there were two chemicals ρ_1 and ρ_2, that react at rate λ and form a third species ρ_3. Suppose that $C_1 + 2C_2 \mapsto C_3$ and that each diffused at their own rates κ_i. Then, the equations would be

$$\partial_t\rho_1 + \mathbf{u}\cdot\nabla\rho_1 = \kappa_1\Delta\rho_1 - \lambda\rho_1\rho_2 \tag{9.6}$$
$$\partial_t\rho_2 + \mathbf{u}\cdot\nabla\rho_2 = \kappa_2\Delta\rho_2 - 2\lambda\rho_1\rho_2 \tag{9.7}$$
$$\partial_t\rho_3 + \mathbf{u}\cdot\nabla\rho_3 = \kappa_3\Delta\rho_3 + \lambda\rho_1\rho_2. \tag{9.8}$$

Exercise 9.1. Show that

$$\frac{d}{dt}\int \rho_1 + \frac{1}{2}\rho_2 + 2\rho_3 d\mathbf{x} = 0. \tag{9.9}$$

This expresses the conservation of total mass as the reactants are changing form.

It makes sense to measure

$$C_3(t) = \int_D \rho_3(t, \mathbf{x}) d\mathbf{x} \tag{9.10}$$

as the total amount of final product produced. The faster C_3 increases, the more efficient the reaction is. We will see in the numerical exercises that different choices of $\mathbf{u}$ can yield different reaction rates. Hence, trying to make intelligent choices for $\mathbf{u}$ can change the effectiveness of various manufacturing jobs, for example.

10 Introduction to fluid mechanics

In all of the above examples, the velocity field is assumed to be known already. However, in many engineering examples (i.e. the design of cars, airplanes, ships, oil extraction and pipeline design) and scientific applications (i.e. weather/climate prediction) the velocity field of the fluid is one of the *unknowns* that must be solved for. To get a hint for what kind of differential equations appear in this context, consider the motion of a particle of fluid, which moves with the fluid velocity field:

$$\frac{d}{dt}\mathbf{X}(t, \alpha) = \mathbf{u}(t, \mathbf{X}(t, \alpha)) \tag{10.1}$$

$$\mathbf{X}(0, \alpha) = \alpha. \tag{10.2}$$

The motion of a particle of fluid will be subject to Newton's equation $ma = F$: mass times acceleration equals force. The acceleration is computed via the chain rule (Theorem B.1):

$$\frac{d^2}{dt^2}\mathbf{X}(t, \alpha) = \frac{d}{dt}\mathbf{u}(t, \mathbf{X}(t, \alpha)) = \partial_t \mathbf{u}(t, \mathbf{X}(t, \alpha)) + \sum_{j=1}^{d} \partial_{x_j}\mathbf{u}(t, \mathbf{X}(t, \alpha))\frac{dX_j}{dt} \tag{10.3}$$

$$= \partial_t \mathbf{u}(t, \mathbf{X}(t, \alpha)) + \sum_{j=1}^{d} \partial_{x_j}\mathbf{u}(t, \mathbf{X}(t, \alpha))u_j(t, \mathbf{X}(t, \alpha)). \tag{10.4}$$

At every point in the fluid, this quantity (times the density of the fluid) will balance the forces acting within the fluid. For reasons which are a little beyond the scope of the course, in the absence of external forces acting on the material, the force inside a fluid (or a solid) is always in the form of a matrix, $\nabla \cdot \sigma$ (called the *Cauchy stress tensor*). The divergence of a matrix is defined row-wise: for example, in 3D

$$\nabla \cdot \sigma = \begin{pmatrix} \partial_{x_1}\sigma_{11} + \partial_{x_2}\sigma_{12} + \partial_{x_3}\sigma_{12} \\ \partial_{x_1}\sigma_{21} + \partial_{x_2}\sigma_{22} + \partial_{x_3}\sigma_{22} \\ \partial_{x_1}\sigma_{31} + \partial_{x_2}\sigma_{32} + \partial_{x_3}\sigma_{32} \end{pmatrix}. \tag{10.5}$$

Therefore, the general equations of motion of a fluid (basically this is just an extension of $ma = F$) is

$$\rho\left(\partial_t \mathbf{u} + \sum_{j=1}^{d} \partial_{x_j}\mathbf{u}u_j\right) = \nabla \cdot \sigma, \tag{10.6}$$

where $\rho(t,x)$ is the density (this may or may not be constant). Pretty much every different type of fluid (or fluid-like substance), has an equation basically this form – it is called a *momentum balance* equation. What distinguishes one fluid from another is the choice of σ. At first, this matrix was derived from experiments and basic physical principles. Now one can also attempt to derive σ from underlying physical principles about molecular dynamics. The simplest type of fluid is a "Newtonian fluid", in which σ is simply the sum of two contributions:

$$\sigma = p(t,x)Id + \mu\frac{D\mathbf{u} + D\mathbf{u}^T}{2} + \tilde{\mu}\nabla\cdot\mathbf{u}Id, \tag{10.7}$$

the scalar function $p(t,x)$ is the *pressure*, Id is the identity matrix, $D\mathbf{u}^T$ denotes the transpose of the Jacobian matrix, and the parameters $\mu, \tilde{\mu} > 0$ are known respectively as the *shear viscosity* and the *bulk viscosity*. The case of incompressible fluid mechanics, that is, where we impose $\nabla\cdot\mathbf{u} = 0$ is easier, and in this case the stress tensor is

$$\sigma = -p(t,x)Id + \mu\frac{D\mathbf{u} + D\mathbf{u}^T}{2}. \tag{10.8}$$

Therefore, the governing equations are the infamous *Navier-Stokes equations*

$$\rho\left(\partial_t\mathbf{u} + \sum_{j=1}^{d}\partial_{x_j}\mathbf{u}u_j\right) = -\nabla p + \mu\Delta\mathbf{u} \tag{10.9}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{10.10}$$

where we performed the following calculation and used the incompressibility again

$$\left[\nabla\cdot\left(D\mathbf{u} + D\mathbf{u}^T\right)\right]_j = \sum_{k=1}^{d}\partial_{x_k}\left(\partial_{x_k}u_j + \partial_{x_j}u_k\right) \tag{10.11}$$

$$= \Delta u_j + \partial_{x_j}\nabla\cdot u \tag{10.12}$$

$$= \Delta u_j. \tag{10.13}$$

Written component-wise the Navier-Stokes equations become

$$\rho\left(\partial_t u_k + \sum_{j=1}^{d}u_j\partial_{x_j}u_k\right) = -\partial_{x_k}p + \mu\Delta u_k \tag{10.14}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{10.15}$$

In incompressible fluid mechanics, it is usually reasonable to assume that ρ is a constant, i.e. *does not depend on* t, x. Hence, replacing $q = \frac{1}{\rho}p$ and $\nu = \frac{\mu}{\rho}$, we further reduce

$$\partial_t u_k + \sum_{j=1}^{d}u_j\partial_{x_j}u_k = -\partial_{x_k}q + \nu\Delta u_k \tag{10.16}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{10.17}$$

There is one thing left: how does one determine the pressure? The pressure is actually determined by the incompressibility condition $\nabla\cdot\mathbf{u} = 0$. Taking ∂_{x_k} of the first equation and summing them gives

$$\sum_{j=1}^{d}\sum_{k=1}^{d}\partial_{x_k}u_j\partial_{x_j}u_k = -\Delta q. \tag{10.18}$$

Hence, the pressure is determined by solving yet another PDE – called Poisson's equation. The equations are hence

$$\partial_t u_k + \sum_{j=1}^{d} u_j \partial_{x_j} u_k = -\partial_{x_k} q + \nu \Delta u_k \tag{10.19}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{10.20}$$

$$\sum_{j=1}^{d} \sum_{k=1}^{d} \partial_{x_k} u_j \partial_{x_j} u_k = -\Delta q, \tag{10.21}$$

though the last equation is usually left off because it is derived from the first two. These equations form the basis of fluid mechanics, i.e. essentially all models in fluid and gas dynamics are some variation of these equations, or include these equations a subset of the mathematical model (though many models are not incompressible and so have different equations for the pressure!). In this way, these fundamental equations underlie the design of ships, airplanes, and cars. They underlie our understanding of how blood moves in our veins, how air moves in our lungs. They underlie our understanding of weather, climate, and the currents in the oceans. They are a pretty important set of equations!

A Taylor's theorem and finite differences

It is convenient to introduce "Big-Oh" notation:

Definition 3 (Big-Oh)**.** Given two functions f, g we say $f = O(g)$ (pronounced f is Big-Oh of g) if there exists a constant such that

$$f(x) \leq Cg(x). \tag{A.1}$$

We will write $f_1 = f_2 + O(g)$ (and similar expressions) if $|f_1 - f_2| = O(g)$.

We recall Taylor's theorem in one dimension, which is kind of an ultimate theorem about local approximation of functions.

Theorem A.1 (Taylor's theorem)**.** *Let f be $k+1$-times differentiable. Then, (using the notation $f^{(k)}$ to denote k derivatives of f),*

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f'''(x) + ... + \frac{h^k}{k!} f^{(k)}(x) + O(h^{k+1}) \tag{A.2}$$

$$= \sum_{j=0}^{k} \frac{h^j}{j!} f^{(j)}(x) + O(h^{k+1}). \tag{A.3}$$

The following *finite-difference approximations* are good exercises in using Taylor's theorem.

Theorem A.2. *Let f be at least three times differentiable. Then*

$$\frac{f(x+h) - f(x)}{h} = f'(x) + O(h) \tag{A.4}$$

$$\frac{f(x) - f(x-h)}{h} = f'(x) + O(h) \tag{A.5}$$

$$\frac{f(x+h) - f(x-h)}{2h} = f'(x) + O(h^2) \tag{A.6}$$

$$\frac{f(x+h) - 2f(x) + f(x-h)}{h^2} = f''(x) + O(h^2). \tag{A.7}$$

Higher order approximations are possible, as are more general forms. However, these will be sufficient for our purposes here.

There is a Taylor's theorem that work in higher dimensions as well, but it starts to become rather tedious to state. It will suffice for us here to just use the following.

Theorem A.3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice differentiable. Then,*

$$f(\mathbf{x}+\mathbf{v}) = f(\mathbf{x}) + \nabla f(\mathbf{x}) \cdot \mathbf{v} + O(\|\mathbf{v}\|^2). \tag{A.8}$$

Let $\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^n$ be twice differentiable. Then,

$$\mathbf{F}(\mathbf{x}+\mathbf{v}) = \mathbf{F}(\mathbf{x}) + D\mathbf{F}(\mathbf{x})\mathbf{v} + O(\|\mathbf{v}\|^2). \tag{A.9}$$

B Vector calculus review or crash-course

We use bold-face $\mathbf{v}$ to denote a vector and plain text v to denote a real number. We denote the set of d-dimensional vectors $\mathbb{R}^d$, that is the set of all vectors

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_d \end{pmatrix}. \tag{B.1}$$

Often, $\mathbf{v}$ can denote a position or velocity (in which case we usually have $d = 1, 2$ or 3) but vectors can be used to store many kinds of information. For example, each v_j could be the stock price of a company in the S & P 500, in which case one would have $d = 500$. In the examples we consider here, we will focus on examples with $d = 1$, 2, or 3. The length of a vector is denoted

$$\|\mathbf{v}\| = \sqrt{v_1^2 + v_2^2 + ... + v_d^2} = \sqrt{\sum_{i=1}^{d} v_i^2}. \tag{B.2}$$

If $\mathbf{v}$ is a position in space, then $\|\mathbf{v}\|$ is the distance to the origin. If $\mathbf{v}$, $\mathbf{u}$ are two positions, then $\|\mathbf{v} - \mathbf{u}\|$ the distance between these two points. We denote e_j the "canonical unit vectors", i.e. the vectors pointed in the cardinal directions:

$$\mathbf{e}_1 = (1, 0, 0, ..., 0) \tag{B.3}$$
$$\mathbf{e}_2 = (0, 1, 0, ..., 0) \tag{B.4}$$
$$\vdots \tag{B.5}$$
$$\mathbf{e}_d = (0, 0, ...1). \tag{B.6}$$

We say any vector $\mathbf{v}$ is a "unit vector" if $\|\mathbf{v}\| = 1$. The dot-product of two vectors is defined as

$$\mathbf{u} \cdot \mathbf{v} = v_1 u_1 + v_2 u_2 + ... + v_d u_d = \sum_{i=1}^{d} v_i u_i. \tag{B.7}$$

Notice that

$$\|v\| = \sqrt{\mathbf{v} \cdot \mathbf{v}}. \tag{B.8}$$

Two vectors $\mathbf{u}, \mathbf{v}$ are called *orthogonal* if $\mathbf{u} \cdot \mathbf{v} = 0$. This is the same thing as being at right angles. Note $\mathbf{e}_j \cdot \mathbf{e}_i = 0$ if $i \neq j$ and so the collection $\{\mathbf{e}_1, ..., \mathbf{e}_d\}$ is a set of orthogonal, unit vectors (called *orthonormal vectors*).

We denote scalar functions $f : \mathbb{R}^d \to \mathbb{R}$ with plain text and vector-fields $\mathbf{F} : \mathbb{R}^d \to \mathbb{R}^d$ with bold-face. A scalar function $f(\mathbf{x})$ could measure, for example, the temperature in the room as a function of $\mathbf{x} \in \mathbb{R}^d$. A vector-field $\mathbf{F}(\mathbf{x})$ could measure, for example, the velocity of the air in the room a function of $\mathbf{x} \in \mathbb{R}^d$. Unless we mention otherwise, we use the notation

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} F_1(\mathbf{x}) \\ F_2(\mathbf{x}) \\ \vdots F_d(\mathbf{x}) \end{pmatrix}. \tag{B.9}$$

Definition 4. A function $f : \mathbb{R}^d \to \mathbb{R}$ is *differentiable*, if there exists a vector-field, denoted $\nabla f(\mathbf{x})$, such that for all $\mathbf{v} \in \mathbb{R}^d$,

$$\frac{d}{dh} f(\mathbf{x} + h\mathbf{v})|_{h=0} = \lim_{h \to 0} \frac{f(\mathbf{x} + h\mathbf{v}) - f(\mathbf{x})}{h} = \nabla f(\mathbf{x}) \cdot \mathbf{v}. \tag{B.10}$$

The vector field ∇f is called the *gradient*.

The gradient $\nabla f(\mathbf{x})$ gives the direction of fastest increase of a scalar function, i.e. the steepest ascent. The direction $-\nabla f(\mathbf{x})$ is the direction of fastest decrease, that is, the steepest descent. The components of ∇f are called *partial derivatives* and are denoted via:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} \partial_{x_1} f(\mathbf{x}) \\ \partial_{x_2} f(\mathbf{x}) \\ \vdots \\ \partial_{x_d} f(\mathbf{x}) \end{pmatrix}. \tag{B.11}$$

Notice that

$$\nabla f(\mathbf{x}) \cdot \mathbf{e}_j = \partial_{x_j} f(\mathbf{x}) = \frac{d}{dh} f(\mathbf{x} + h\mathbf{e})|_{h=0} = \lim_{h \to 0} \frac{f(\mathbf{x} + h\mathbf{e}_j) - f(\mathbf{x})}{h} \tag{B.12}$$

Definition 5. A vector-field $\mathbf{F} : \mathbb{R}^d \to \mathbb{R}^d$ is called *differentiable* if for all $\mathbf{x}$, there is a matrix $D\mathbf{F} \in \mathbb{R}^{d \times d}$ such that for al $\mathbf{v}, \mathbf{u} \in \mathbb{R}^d$

$$\frac{d}{dh} \mathbf{F}(\mathbf{x} + h\mathbf{v})|_{h=0} = \lim_{h \to 0} \frac{\mathbf{F}(\mathbf{x} + h\mathbf{v}) - \mathbf{F}(\mathbf{x})}{h} = D\mathbf{F}(\mathbf{x})\mathbf{v}. \tag{B.13}$$

the matrix $D\mathbf{F}$ is called the *Jacobian* (named after German mathematician Carl Gustav Jacob Jacobi).

By unwrapping the above annoying definition we see that the i, j-th entry in $D\mathbf{F}$ is given as

$$(D\mathbf{F})_{ij} = \partial_{x_j} F_i. \tag{B.14}$$

For example, in two dimensions

$$D\mathbf{F} = \begin{pmatrix} \partial_{x_1} F_1 & \partial_{x_2} F_1 \\ \partial_{x_1} F_2 & \partial_{x_2} F_2 \end{pmatrix}. \tag{B.15}$$

For vector-fields $\mathbf{F}$ we define the *divergence* (a scalar function),

$$\nabla \cdot \mathbf{F} = \partial_{x_1} F_1 + ... + \partial_{x_d} F_d = \sum_{i=1}^{d} \partial_{x_i} F_i. \tag{B.16}$$

Definition 6. A function $f : \mathbb{R}^d \to \mathbb{R}$ is *twice differentiable* f is differentiable and ∇f is differentiable (as a vector-field).

The second partial derivatives are defined in a similar way:

$$\partial_{x_j}\partial_{x_i} f(\mathbf{x}) = \partial_{x_j x_i} f(\mathbf{x}) = \frac{d}{dh}\partial_{x_i} f(\mathbf{x} + h\mathbf{e})|_{h=0} = \lim_{h\to 0} \frac{\partial_{x_i} f(\mathbf{x} + h\mathbf{e}_j) - \partial_{x_i} f(\mathbf{x})}{h}. \tag{B.17}$$

If f is twice-differentiable then we can have *equality of mixed partials*:

$$\partial_{x_j x_i} f = \partial_{x_i x_j} f. \tag{B.18}$$

The last differential operator we will need is the *Laplacian* (named after French mathematician Pierre-Simon Laplace), for a scalar function f,

$$\Delta f = \nabla \cdot \nabla f = \partial_{x_1 x_1} f + \partial_{x_2 x_2} f + ... + \partial_{x_d x_d} f = \sum_{i=1}^{d} \partial_{x_i x_i} f. \tag{B.19}$$

We will also have examples where scalar functions and vector-fields depend on both time and space, denoted $f(t, \mathbf{x})$ and $\mathbf{F}(t, \mathbf{x})$. In this case $\nabla f(t, \mathbf{x})$ denotes only derivative with respect to space

$$\nabla f(t, \mathbf{x}) = \begin{pmatrix} \partial_{x_1} f(t, \mathbf{x}) \\ \partial_{x_2} f(t, \mathbf{x}) \\ \vdots \\ \partial_{x_d} f(t, \mathbf{x}) \end{pmatrix}. \tag{B.20}$$

Similar conventions hold for the Jacobian, divergence, and Laplacian. We denote $\partial_t f$ to be the partial derivative with respect to time:

$$\partial_t f(t, \mathbf{x}) = \lim_{h\to 0} \frac{f(t + h, \mathbf{x}) - f(\mathbf{x})}{h}. \tag{B.21}$$

Sometimes we will use the multi-dimensional chain rule.

Theorem B.1 (The chain rule)**.** *Let $f(t, \mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ and $\mathbf{G}(t, \mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^d$ both be differentiable. Then,*

$$\partial_{x_j} \left(f(t, \mathbf{G}(t, \mathbf{x})) \right) = \sum_{i=1}^{d} \partial_{x_i} f(t, \mathbf{G}(\mathbf{x})) \partial_{x_j} G_i(t, \mathbf{x}). \tag{B.22}$$

and

$$\partial_t \left(f(t, \mathbf{G}(t, \mathbf{x})) \right) = \partial_t f(t, \mathbf{G}(t, \mathbf{x}) + \sum_{i=1}^{d} \partial_{x_i} f(t, \mathbf{G}(\mathbf{x})) \partial_t G_i(t, \mathbf{x}). \tag{B.23}$$

When integrating scalar functions (or vector-fields) over rectangles and squares, we can define the volume or area integral as a multiple integral of one-dimensional integrals. For example, we can denote the domain $D = [a, b] \times [c, d]$ the rectangle in $\mathbb{R}^2$ with corners $(a, c), (b, c), (b, d), (a, d)$ (labeled counter-clockwise) and then the area integral of a scalar function $f(\mathbf{x})$ over D is defined as

$$\int_D f(\mathbf{x}) d\mathbf{x} = \int_c^d \int_a^b f(x_1, x_2) dx_1 dx_2 = \int_a^b \int_c^d f(x_1, x_2) dx_2 dx_1. \tag{B.24}$$

Note that we can order the integrals in whatever order we want. Given any 'nice' domain $D \subset \mathbb{R}^d$, we denote ∂D the boundary of the domain. For the above rectangle for example, it is the 4 lines connecting the corners; by convention they are listed in counter-clockwise order. Most domains D that you encounter in real life all have perfectly reasonable notions of boundary that are evident. For example, if D is a polygon, the boundary consists of the edges, the set of vectors in $\mathbb{R}^3$ with $\|\mathbf{x}\| \leq 1$, boundary is the sphere of radius 1. Basically any reasonable domain will have a reasonable ∂D and we can define the multi-dimensional integral $\int_D f(\mathbf{x}) d\mathbf{x}$ by taking the limit of smaller and smaller boxes (similar to how the Riemann integral is defined in one dimension). We will only have to deal with circles, spheres, and polygons here. For all such domains, for any $\mathbf{x} \in \partial D$ we can similarly define $\mathbf{n}(\mathbf{x})$ the *outward unit normal*, which is the unit vector that is orthogonal to the boundary and which points *out* of the boundary. For example, consider the disk of radius $R > 0$, $D_R = \{\mathbf{x} \in \mathbb{R}^2 \text{ such that } \|\mathbf{x}\| \leq R\}$. Then, $\partial D_R = \{\mathbf{x} \in \mathbb{R}^2 \text{ such that } \|\mathbf{x}\| = R\}$. Notice that if $\mathbf{x} \in \partial D_R$, then there is a $\theta \in [0, 2\pi)$ such that $\mathbf{x} = R \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$. Conveniently, if $\mathbf{x} \in \partial D_R$ then we have $\mathbf{n}(\mathbf{x}) = \frac{1}{R}\mathbf{x}$. For nice domains we can also define a *surface integral*

$$\int_{\partial D} f(\mathbf{x}) dS(\mathbf{x}). \tag{B.25}$$

If D is a two-dimensional area, then the surface integral is defined as an integral along the lines/curve making up the boundary of D. If D is a three-dimensional volume, then it is a multi-dimensional integral over the areas. For all but very simple examples, doing actual calculations of surface integrals by hand is very painstaking and annoying. We won't need to do anything of the sort.

For nice domains, we have the all-important *divergence theorem* that relates most of the above concepts: If $\mathbf{F}$ is a differentiable vector field and D is a bounded domain in $\mathbb{R}^d$, then

$$\int_D \nabla \cdot \mathbf{F} d\mathbf{x} = \int_{\partial D} \mathbf{F} \cdot \mathbf{n}(\mathbf{x}) dS(\mathbf{x}). \tag{B.26}$$

You should think of this as the multi-dimensional version of the fundamental theorem of calculus:

$$\int_a^b f(x) dx = f(b) - f(a). \tag{B.27}$$

C Eigenvalues and eigenvectors

Let $\mathbb{R}^{n \times n}$ denote n-by-n matrices with real entries. Recall matrix multiplication of vectors $\mathbf{v} \in \mathbb{R}^n$ (or $\mathbf{v} \in \mathbb{C}^n$) is defined as

$$(A\mathbf{v})_k = \sum_{j=1}^{n} A_{kj} v_j. \tag{C.1}$$

We denote $I \in \mathbb{R}^{n \times n}$ *the identity matrix*

$$I = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ & \vdots & & \vdots & \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}. \tag{C.2}$$

It is the unique matrix for which $I\mathbf{v} = \mathbf{v}$ for all $\mathbf{v} \in \mathbb{R}^n$. Matrix-matrix multiplication AB is defined by applying A successively to the columns of B:

$$(AB)_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}. \tag{C.3}$$

Definition 7. Let $A \in \mathbb{R}^{n\times n}$ be an n-by-n matrix with real entries. We call $\lambda \in \mathbb{C}$ and $\mathbf{v} \in \mathbb{C}^n$ an eigenvalue/eigenvector pair if

$$A\mathbf{v} = \lambda\mathbf{v}. \tag{C.4}$$

Notice that an eigenvalue/eigenvector pair solves

$$(A - \lambda I)\,\mathbf{v} = 0. \tag{C.5}$$

Hence, the set of all eigenvalues is given by $\det(A - \lambda I) = 0$. If one recalls the definition of determinant, one sees that eigenvalues are the roots of an n-degree polynomial. By the fundamental theorem of algebra, this means there are exactly n eigenvalues, counting multiplicity. For example, for the identity matrix I, the polynomial is

$$(1 - \lambda)^n = 0, \tag{C.6}$$

which has only the root $\lambda = 1$, but it has multiplicity n.

We will only discuss the example of $\mathbb{R}^{2\times 2}$. If we have a matrix

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tag{C.7}$$

then $\det(A - \lambda I) = 0$ is the polynomial equation

$$(a - \lambda)(d - \lambda) - bc = 0. \tag{C.8}$$

Once one finds eigenvalues, one then would try to solve (C.5) to find corresponding eigenvectors.

Definition 8 (Real diagonalizability)**.** A matrix A is called *real diagonalizable* if all of the eigenvalues $\lambda_1, ..., \lambda_n$ (with some possible repetitions) are real and there is an invertible matrix $P \in \mathbb{R}^{n\times n}$ such that

$$A = P \begin{pmatrix} \lambda_1 & 0 & \dots & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ & \vdots & & \vdots & \\ 0 & \dots & \dots & 0 & \lambda_n \end{pmatrix} P^{-1} \tag{C.9}$$

Note that if one considers the formula

$$AP = P \begin{pmatrix} \lambda_1 & 0 & \dots & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ & \vdots & & \vdots & \\ 0 & \dots & \dots & 0 & \lambda_n \end{pmatrix}, \tag{C.10}$$

we see that the j-th column of P is an eigenvector corresponding to λ_j. Not every matrix with real eigenvalues is real diagonalizable. Consider for example

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}. \tag{C.11}$$

This matrix has two zero eigenvalues but is not zero; and so it cannot be real diagonalizable.

Not every real matrix has real eigenvalues, for example the matrix

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \tag{C.12}$$

as eigenvalues which solve

$$\lambda^2 + 1 = 0, \tag{C.13}$$

i.e. $\lambda_\pm = \pm\sqrt{-1}$. In this case, one can in general diagonalize the matrix, but P and the λ_j's will be complex. There is a different kind of matrix decomposition that is more convenient, at least for $\mathbb{R}^{2\times 2}$ matrices.

Theorem C.1. *Let $A \in \mathbb{R}^{2\times 2}$ with eigenvalues $\lambda_\pm = \lambda_r \pm i\lambda_i$ where $\lambda_r, \lambda_i \in \mathbb{R}$ and $\lambda_i \neq 0$. Then, there is an invertible matrix $P \in \mathbb{R}^{2\times 2}$ such that*

$$A = P \begin{pmatrix} \lambda_r & \lambda_i \\ -\lambda_i & \lambda_r \end{pmatrix} P^{-1}. \tag{C.14}$$

The above examples (real diagonalizable, non-diagonalizable, and complex diagonalizable) do essentially exhaust the possibilities however: all matrices can, up to a change of variables, be placed in essentially some combination of the three above cases.

References

[BN12] Fred Brauer and John A Nohel. *The qualitative theory of ordinary differential equations: an introduction*. Courier Corporation, 2012.

[Ise09] Arieh Iserles. *A first course in the numerical analysis of differential equations*. Number 44. Cambridge university press, 2009.

[Str04] John C Strikwerda. *Finite difference schemes and partial differential equations*, volume 88. Siam, 2004.