Modelling with ordinary differential equations III

Dr Luke Bennetts
School of Mathematical Sciences
mailto:luke.bennetts@adelaide.edu.au

Semester 1, 2018

Contents

1 Introduction

1.1 Preliminaries

This course is concerned with differential equations (DEs) having just one *independent variable*, which are known as ordinary differential equations (ODEs). The course Partial Differential Equations and Waves (Semester 2) tackles differential equations with multiple independent variables. We will consider scalar ODEs having a single *dependent variable* and vector ODEs, equivalently systems of ODEs, having vector (or multiple) dependent variables.

1.2 Dynamics, modelling and computation

Dynamics, modelling and computation are the the key ingredients of this course.

1.2.1 Dynamics

Dynamics refers to the *temporal structure* of a quantity.

Typically, we seek the behaviour of a function $u(t)$, which describes some physical quantity u (location, temperature, population, etc), as a function of time t. To do this we will use an ODE with one or more *initial conditions* (ICs), giving an *initial value problem* (IVP).

Alternatively, we may seek the *spatial structure* of a quantity, i.e. $u(x)$, where x denotes location. For this we need an ODE with one or more boundary conditions, and the problem will be a *boundary value problem* (BVP).

1.2.2 Modelling

Mathematical modelling is the process of deriving equations for real-word problems, with associated initial and/or boundary conditions.

Differential equations model an enormous range of phenomena in many applications areas in physics, chemistry, biology, sociology, engineering and games. We 'solve' differential equations to predict and understand behaviours and complex interactions in such applications.

1.2.3 Computation

Ideally, we would like to calculate an *exact solution* to a given ODE, as exact solutions contain no error. However, for most interesting ODEs we can't find exact solutions. In fact, we don't necessarily know if a solution exists, or if solutions are unique!

If we do know that a unique solution exists, then we can compute an approximate solution, which contains some error. The amount of error that can be tolerated depends on the application. In general there is a trade-off between computational error and computational time. You should choose a method that gives the required accuracy without wasting time and resources. If you want a 'ball park' answer, use a fast method with low-order accuracy; if you need a very accurate answer you will need to use a slower method with high-order accuracy. Remember that mathematical models themselves are approximations of the real world!

1.2.4 Analysis

Analysis underpins all aspects of dynamics, modelling and computation.

For an applied mathematician, analysis refers to finding general patterns and understanding what they tell us. Analysis can be in terms of the mathematical properties of the ODE alone, but it is essential to analyse the properties of the ODE in order to interpret the what the mathematical model predicts for the real-world problem we began with.

1.3 Examples

1.3.1 Uniqueness of solution; well and ill posed problems

Consider the first-order initial value problem

$$\dot{x} = x^{1/3}, \quad x(0) = 0. \tag{1.1}$$

We assume that solutions of interest are real; complex numbers significantly complicate matters!

Clearly, we have the trivial solution $x = 0$. Is this the only solution? No!

The existence and uniqueness of model solutions is something often taken for granted. We will look at Picard's existence and uniqueness theorem later. For now we note that lack of these things means that in devising the model we must have left out some important feature of the problem, i.e. the problem is *ill posed*.

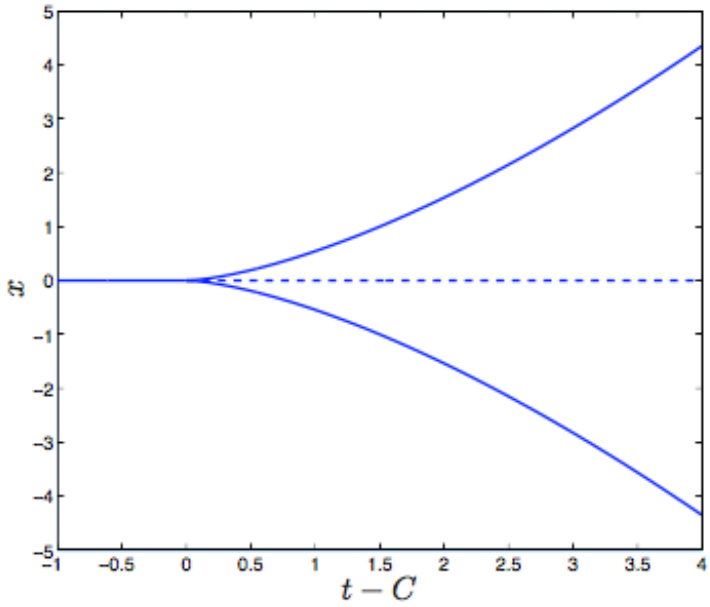

Fig. 1.1: Solutions of (1.1).

If the initial condition for (1.1) was $x(0) = 1$ (or any other positive value) then there would be a unique, non-trivial solution:

$$x(t) = \begin{cases} 0, & t < -\frac{3}{2}, \\ +\sqrt{\frac{8}{27}\left(t + \frac{3}{2}\right)^3}, & t \geq -\frac{3}{2}, \end{cases} \tag{1.2}$$

where we must take the positive square root. The problem is now *well posed*. We may or may not care about the part of the solution prior to $t = 0$. If the initial condition was $x(0) = -1$ then we'd take the negative square root.

What would happen if we mindlessly decided to solve (1.1) numerically using Euler's method?

Well and ill posed problems

The concept of well and ill posed problems was defined by Jacques Hadamard (1865–1963). He believed that for mathematical models of physical phenomena:

1. A solution must exist.
2. The solution must be unique.
3. The solution's behaviour must change continuously with the initial conditions.

A problem that is well posed stands a good chance of numerical solution; one that is ill posed does not and will need to be reformulated.

Fig. 1.2: Jacques Hadamard (1865–1963).
http://en.wikipedia.org/wiki/Jacques_Hadamard

1.3.2 Importance of theory and numerical accuracy

Consider the free-surface profile

$$z = \delta\, h(y),\ -1 \le y \le 1,\ \delta \ll 1, \tag{1.3}$$

for a thin a layer of fluid flowing down a rectangular channel of width 2, wound helically about a vertical axis. An ODE model for the depth h is

$$\frac{dh}{dy} = \frac{h^4}{1+\epsilon\, y}, \tag{1.4}$$

where ϵ is the channel curvature and $0 < \epsilon < 1$, subject to the boundary condition

$$h(-1) = h_\ell. \tag{1.5}$$

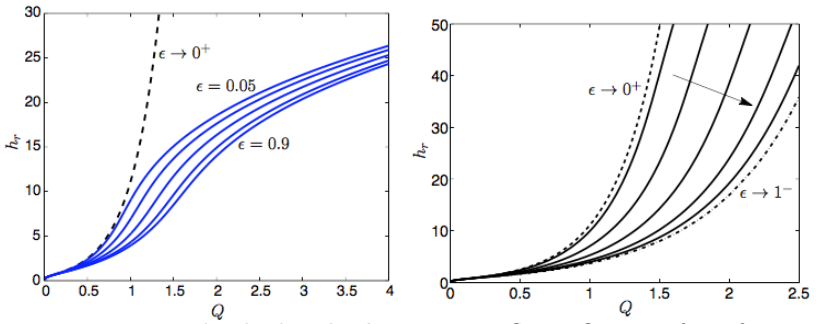

Fig. 1.3: Fluid depth h_r versus flux Q. Left: for $\epsilon = 0.05, 0.25, 0.5, 0.75, 0.9$ (solid), calculated using trapezoidal rule (`trapz|` in Matlab) in and in asymptotic limit $\epsilon \to 0^+$ (dashed). Right: numerical computations use Simpson's rule or Matlab's `quad`, `quad1`, `quadgk` (all more accurate than the trapezoidal rule) and the same plot was obtained in each case. Asymptotic limit $\epsilon \to 1^-$ (dash-dot) also included.

We can calculate an exact solution to this ODE. We then wish to compute the flux down the channel

$$Q = \frac{1}{3}\int_{-1}^{1} h^3\, dy. \tag{1.6}$$

In general we must determine Q by numerical quadrature. However, in the limits of zero and unit curvature we can get exact relations.

Exact and numerical solutions

Figure 1.3 shows the fluid depth h_r at the right boundary $y = 1$ versus flux Q for different values of ϵ. The top panel shows the result when the trapezoidal rule was used to compute Q (1.6) numerically for $0 < \epsilon < 1$. The bottom panel shows the result when more accurate quadrature rules were used to compute Q. Determining the solution in the unit curvature limit indicated numerical inaccuracy and prompted use of more accurate quadrature rules. Computational methods were necessary, but the accuracy of the method was important as Q became large and h_r became large and the exact solutions were critical to us finding this.

We may write

$$\frac{1}{h_\ell^3} - \frac{1}{h_r^3} = \eta \quad \text{where} \quad \eta = \frac{3}{\epsilon}\log\left(\frac{1+\epsilon}{1-\epsilon}\right), \tag{1.7}$$

from which we see that for physically meaningful (i.e., finite and positive) h_r we require $h_\ell < \eta^{-1/3}$. This expression also shows that $h_r \to \infty$ and $dh/dy \to \infty$ as $h_\ell \to \eta^{-1/3}$, i.e., both the fluid depth and the free surface slope at the right-hand boundary become infinite in this limit. This is very useful information because it tells us that we will have trouble finding a solution if we choose h_ℓ too large. We might easily do this if we were to just solve the problem numerically.

2 One-dimensional autonomous ODE models

2.1 Autonomous scalar equations

A *first-order scalar ODE* is of the form

$$\frac{dx}{dt} = f(x(t), t).$$

If the function f does not depend explicitly on t, i.e.

$$\frac{dx}{dt} = f(x). \tag{2.1}$$

then the equation is *autonomous*. For such equations, *phase-line analysis* determines the *qualitative* behaviour of solutions without having to calculate the solution $x(t)$.

2.2 Fixed points, phase line analysis, and stability criteria

The *steady states*, *fixed points* or *equilibria* of (2.1), are the values of x such that

$$\frac{dx}{dt} = 0 \qquad \Leftrightarrow \qquad f(x) = 0.$$

Consider the logistic growth equation

$$\frac{dx}{dt} = f(x), \quad f(x) = rx\left(1 - \frac{x}{k}\right), \tag{2.2}$$

which models population size x. Here rx is the growth term (growth proportional to population size, r = growth rate), rx^2/k is the death term due to competition between in the population, k is the carrying capacity of the population domain. The right-hand side $f(x)$ is plotted in Fig. 2.1. The fixed points are $x = 0$ and $x = k$.

Suppose that the system is at one of the fixed points, which we shall denote x^*. If the system is given a small 'kick', i.e. perturbed slightly, so that it moves a little away from $x = x^*$ then the sign of $f(x)$ tells us the direction of travel from the perturbed state. If $f(x) > 0$ then x increases with time, as shown by an arrow to the right. Conversely, if $f(x) < 0$ then x decreases with time, as shown by an arrow to the left.

From the plot, we see that if the system moves slightly from $x = 0$, it will continue to move away from this fixed point. The fixed point $x = 0$ is said to be *unstable*. If the system moves slightly from $x = c$

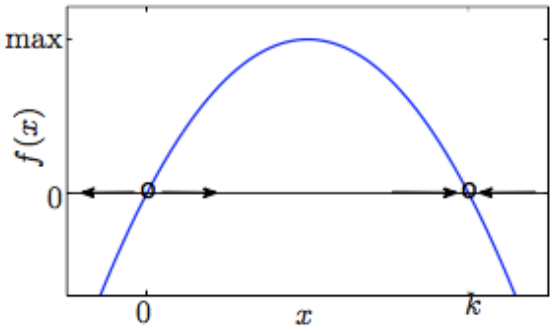

Fig. 2.1: Phase-line analysis of the logistic equation. The fixed points are denoted 'o' and the arrows show the direction of travel if x is perturbed away from a fixed point.

it will move back to this fixed point. The fixed point $x = c$ is said to be *stable*.

This is an easy, *graphical* way to determine the stability of fixed points. We need only to plot the right-hand side of the ODE (2.1) and consider its sign either side of the fixed points $f(x) = 0$. This gives *qualitative* information about the solution.

Sometimes we might want a more *quantitative* way of establishing stability, e.g. if we were using a computer program for this. Again, let x^* be a steady state of (2.1), and suppose that the system is initially at $x = x^*$, but then perturbed to $x^* + \delta x$, then

$$\dot{x}(x^* + \delta x) \approx \delta x f'(x^*), \tag{2.3}$$

where primes denote differentiation with respect to x.

- If $f'(x^*) < 0$, then x is decreasing for $\delta x > 0$ and increasing for $\delta x < 0$ — i.e. the system moves back towards x^* and the steady state $x = x^*$ is a stable.
- If $f'(x^*) > 0$ then x is increasing for $\delta x > 0$ and decreasing for $\delta x < 0$, i.e. the system moves further away from x^* and the steady state is unstable.
- If $f'(x^*) = 0$, i.e. x^* is a stationary point of $f(x)$, we need to consider the next order non-zero term in our expansion. There are three possibilities:
  1. in the neighbourhood of $x = x^*$, $f(x)$ is a decreasing function of x so that the fixed point is stable;
  2. in the neighbourhood of $x = x^*$, $f(x)$ is an increasing function of x so that the fixed point is unstable;
  3. $x = x^*$ is a turning point of $f(x)$ and the fixed point is *semi stable* or *half stable* because the direction of travel is towards the fixed point from one side and away from it on the other side. Because we can never determine

which way a system will be perturbed, we must assume that such a point is, essentially, unstable.

From this we conclude that:

$$\boxed{\text{A fixed point } x = x^* \text{ is stable if } \quad f'(x^*) < 0.}$$

Example Find the fixed points of the following autonomous differential equations, and determine their stability:

(a) $\dfrac{dx}{dt} = x^2$,

(b) $\dfrac{dx}{dt} = x(1-x)(2-x)$.

Plotting the *vector field* gives an even better qualitative understanding of the solution:

- Evaluate the slope of the solution $x(t)$ at many points (t, x) – given by $f(x)$.
- At each point (t, x) draw a short arrow indicating the slope at that point.
- Solution curves are tangential to these short arrows.

Vector fields are readily plotted using Matlab's 'quiver' command.

2.3 Fisheries management; saddle-node and transcritical bifurcations

As demand for fish and seafood grows worldwide, it is becoming increasingly important that fisheries are managed so as to avoid over-fishing, which could lead to catastrophic disruption of the ecosystem, and total collapse of fish numbers. Mathematical models allow fisheries managers to understand the dynamics of the fish populations, and the impact of various management strategies upon them.

Consider an isolated population of fish. What is the best fishing strategy to maximise the yield, whilst also maintaining the population? We explore this using the model

$$\frac{dN}{dt} = \underbrace{BN}_{birth} - \underbrace{DN^2}_{death} - \underbrace{g(N)}_{yield}, \tag{2.4}$$

where N is the number of individuals. The 'birth' term tells us that the population increases at a rate proportional to the population size; B is the birth rate per individual. The 'death' term tells us that the population decreases at a rate proportional to the square of the population size, which measures the number of interactions between

individuals in a large well-mixed population; the individuals are all competing for the same resources (in particular food) and the larger the population the more competition and deaths. The 'yield' term tells us the depletion in the fish population per unit time due to fishing; we need to decide on a functional form for this term. If $g(N) = 0$, we have just a logistic growth model.

2.3.1 Nondimensionalisation

The model (2.4) has at least three parameters: the birth and death rates, B and D, a measure of the population size $\mathcal{N}$, and any parameters that arise in the function $g(N)$. It is always a good idea to non-dimensionalise your model for a number of reasons. Let's first do the non-dimensionalisation, which will give a context for discussing the reasons.

Let's define dimensionless variables, denoted by stars:

$$N = \mathcal{N} N_*, \quad t = \mathcal{T} t_*. \tag{2.5}$$

By choosing $\mathcal{T} = 1/B$ and $\mathcal{N} = K$ we reduce the number of parameters in our model, which becomes

$$\frac{dN_*}{dt_*} = N_*\,(1 - N_*) - g_*(N_*). \tag{2.6}$$

In tutorial 1 we consider a constant yield fishing term, $g(N) = Y$. Then

$$g_*(N_*) = \frac{\mathcal{T} Y}{\mathcal{N}} = \frac{Y}{BK} = \frac{YD}{B^2} = y,$$

and we have the dimensionless equation

$$\frac{dN_*}{dt_*} = N_*(1 - N_*) - y \tag{2.7}$$

with just one parameter y.

Note: if $g(N) = 0$ we have a dimensionless logistic equation with no parameters. This one equation gives the solution for all physical logistic-growth problems, but is subject to an initial condition that adds a parameter.

Once we have the solution to our dimensionless problem we need to know the values K and B to convert answers to physical results, $N = KN_*$, $t = t_*/B$.

Scaling or non-dimensionalising equations is a very important part of modelling.

1. Typically, it reduces the number of parameters, which makes analysis easier.
   (a) It is easier to see the type of equation and identify a solution method.

(b) It is preferable to see how a solution changes with parameters if there are fewer parameters. In the example above, we can see how the solution changes with the parameters y and $N_*(0)$, rather than with four parameters B, K, Y and $N(0)$. For the logistic equation there are no parameters, so we just have to find the solution to one equation and we effectively have the solution for all logistic-growth problems in terms of the initial condition parameter. This is especially useful if we have to find the solution numerically, all the more so if the simulation takes a long time.

(c) Results can be presented more compactly — we can plot N versus t for different values of y or $N(0)$ at a given final time t_f versus y.

(d) Solutions obtained for one system can be applied to another with different parameter values but which obeys the same scaled equation - there is no need to recalculate the solution.

2. It aids experimental validation of a model using a smaller-scale version; from the dimensionless equations we can see how to choose parameters for an experiment while not changing the behaviour of the model.

3. Whether the number of parameters are reduced or not, we can see the relative importance of the different terms in the equation(s) and, perhaps, use this to simplify the equation(s) to be solved. For example, if y in (2.7) is a very small number, say $O(\epsilon) \ll 1$, then we can drop it and conclude that the rate of fishing is so small relative to the growth rate of the population that fishing is unimportant for determining the population size.

Let's discuss reason 3 more carefully. For a constant yield term, our scaled equation is

$$\frac{dN_*}{dt_*} = (\mathcal{T}B)N_* - \left(\frac{\mathcal{T}B\mathcal{N}}{K}\right)N_*^2 - \frac{\mathcal{T}Y}{\mathcal{N}}. \tag{2.8}$$

Concluding that the constant term can be dropped from the equation is dependent on the scales we are using for $\mathcal{T}$ and $\mathcal{N}$ so we need to be a little careful. If the population size is $O(K)$ and the time scale of interest to us is $O(1/B)$ and y is small, then it makes sense to drop it. But, what if the time scale of interest is $\mathcal{T} \ll 1/B$ or if the initial population size is $N(0) = N_0 \ll K$? Then the important terms in the equation might be different. If $\mathcal{T} = 1/B$ but $N_0 \ll K$, then we should choose $\mathcal{N} = N_0$ we have

$$\frac{dN_*}{dt_*} = N_*\left(1 - \frac{N_0}{K}N_*\right) - \frac{Y}{BN_0}. \tag{2.9}$$

Since $N_0/K \ll 1$ we might drop this term. We should only drop the constant term if $Y/(BN_0)$ is small.

What would we do if $\mathcal{T} \ll 1/B$?

This shows that we need to choose scales appropriately based on the information we have and what we wish to investigate.

2.3.2 Constant yield; saddle-node bifurcation

Tutorial 1 shows that the model (2.7) has a *bifurcation* at $y = 1/4$. For $y > 1/4$ there were no (real) steady states, for $y = 1/4$ there was one and for $y < 1/4$ there were two. It was called a *saddle-node bifurcation*. This is the basic mechanism by which fixed points are *created and destroyed*. As a parameter (here y) is varied, two fixed points move toward each other, collide, and mutually annihilate.

More generally, if a small variation of a parameter causes a profound change in the qualitative behaviour of the solution, we call it a *bifurcation*.

Definition: For the ODE $\dot{x} = f(x, \mu)$, where μ is a scalar parameter, $\bar{x}$ is a bifurcation point and $\bar{\mu}$ is a bifurcation value if

$$f(\bar{x}, \bar{\mu}) = 0 \quad \text{and} \quad \frac{\partial f}{\partial x}(\bar{x}, \bar{\mu}) = 0, \tag{2.10}$$

where $\frac{\partial f}{\partial x}$ denotes the partial derivative with respect to x.

Note that for (2.7) ($\dot{N}_* = N_*(1 - N_*) - y$) we find $f(N_*, y) = 0$ and $\partial f/\partial N_* = 1 - 2N_* = 0$ for $N_* = 1/2$ and $y = 1/4$; we have a bifurcation.

The canonical form for a saddle-node bifurcation is the first order ODE

$$\dot{x} = \mu - x^2. \tag{2.11}$$

The right-hand side is just the simplest quadratic polynomial with a constant parameter. In (2.7) we have a slightly more complicated quadratic polynomial.

Note: we are using the standard notation $\dot{x} \equiv dx/dt$.

The right-hand-side of (2.11) has no real roots for $\mu < 0$, one ($x = 0$) for $\mu = 0$ and two ($x = \pm\sqrt{\mu}$) for $\mu > 0$ (see Fig. 2.2). We have a saddle-node bifurcation as μ increases from negative to positive.

For $\mu > 0$ which root is stable and which is unstable?

Once we have determined the stability of the fixed point as functions of μ, we plot the *bifurcation diagram* shown in Fig. 2.3, which shows the roots $x = \pm\sqrt{\mu}$ versus μ, with stability indicated on the two *branches*.

A saddle-node bifurcation is sometimes called a *fold bifurcation* or a *turning point bifurcation*. The term "saddle-node" derives from an analogous bifurcation in higher-dimensional space.

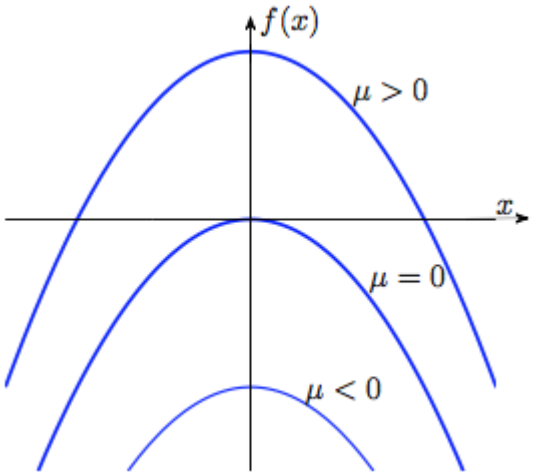

Fig. 2.2: Roots x_* of $f(x) = \mu - x^2$: $x_* = \pm\sqrt{\mu}$ for $\mu > 0$, $x_* = 0$ for $\mu = 0$ and none at all for $\mu < 0$.

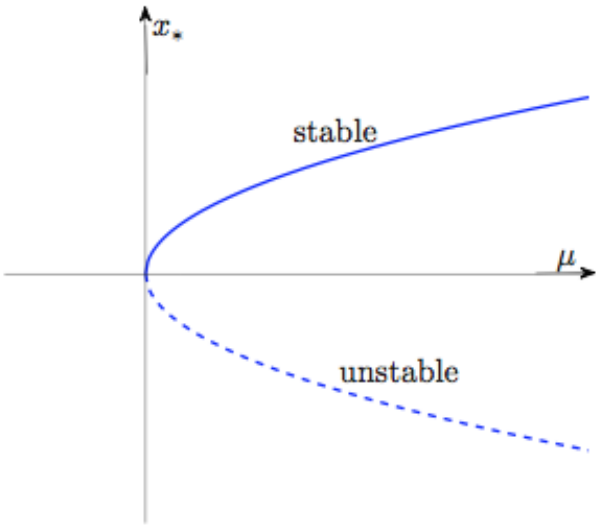

Fig. 2.3: The bifurfaction diagram for $f = \mu - x^2$, showing the fixed points x_* versus parameter μ, with stability indicated.

2.3.3 Constant effort model; transcritical bifurcation

We now consider the model with a "constant effort" fishing term:

$$\frac{dN}{dt} = \underbrace{BN}_{birth} - \underbrace{DN^2}_{death} - \underbrace{EN}_{fishing} . \tag{2.12}$$

The parameter E is the effort put into fishing; the higher the effort, the more fish are caught per unit time. Thus effort includes factors such as the number of fishermen, boats, their working hours, etc.

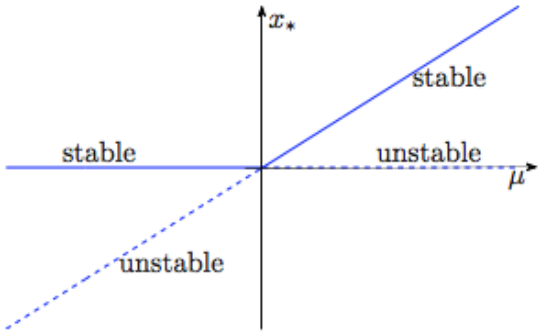

Fig. 2.4: Bifurcation diagram for the transcritical bifurcation.

The scaled or dimensionless model, using $\mathcal{T} = 1/B$, $K = B/D$, $\mathcal{N} = K$, is:

$$\frac{dN_*}{dt_*} = N_*\,(1 - N_*) - eN_*, \quad e = \frac{E}{B}. \tag{2.13}$$

Here we have one parameter e instead of four $(\mathcal{N}, B, D, E)$.

This equation is of the form

$$\dot{x} = \mu x - x^2, \tag{2.14}$$

where $x \equiv N_*$ and $\mu = 1 - e$.

What happens as μ goes from negative to positive?

Eqn. (2.14) is the canonical form for a *transcritical bifurcation*.

With a transcritical bifurcation, two branches of equilibria exchange their stability as μ passes through the bifurcation value (here $\mu = 0$). Fig. 2.4 showns the the bifurcation diagram, i.e. the steady states versus μ.

What does this tell us about model Eqn. (2.13)? In particular, what is the maximum yield that can be achieved?

Fig. 2.5 illustrates the procedure to calculate the maximum non-dimensional yield $Y_{\max}$. What is the dimensional maximum yield?

An alternative scaling

We could have scaled Eqn. (2.13) differently: $\mathcal{T} = 1/(B - E)$, $k = (B - E)/D$, $\mathcal{N} = k$. Then the model equation is just the logistic-growth equation,

$$\frac{dN_*}{dt_*} = N_*\,(1 - N_*)\,, \tag{2.15}$$

with no parameters at all! (Note that we are, implicitly assuming $E < B$ or $k > 0$.) This is nice but how do we examine how the fishing rate affects the yield, now that the fishing rate is involved in the scaling of the problem?

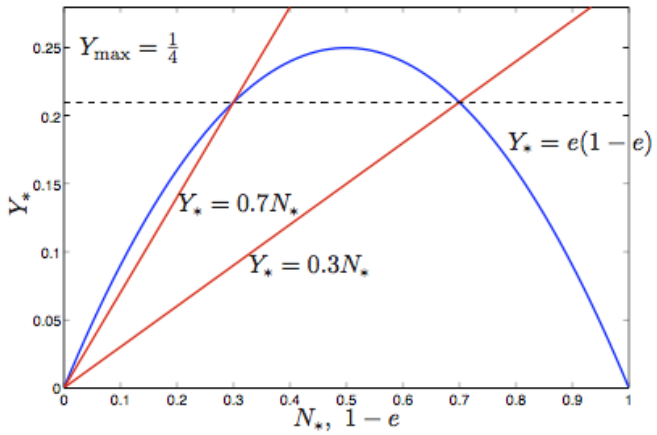

Fig. 2.5: Red: yield $Y_*(N_*) = eN_*$, $e = 0.3, 0.7$. Blue: yield at the non-trivial steady state $N* = 1 - e$, $Y_*(e) = e(1 - e)$.

This example shows that we can scale a problem in different ways. How we scale the problem will affect the way information can be obtained. In this last example we used dimensional, rather than dimensionless yield. It is easier to see how changing the fishing rate changes the yield, using the dimensionless model, if we don't include the fishing rate in the scaling of the problem.

Choosing scaling for a problem and understanding the dimensionless results is not always easy; it takes practice.

2.4 Pitchfork bifurcation; jumps and hysteresis

There is a third kind of bifurcation that can arise in one-dimensional autonomous ODEs, called a *pitchfork bifurcation*. They are common in physical problems that have symmetry. Fixed points tend to appear and disappear in symmetrical pairs. Consider, for example, a thin sheet that is being compressed, as illustrated in Fig. 2.6.

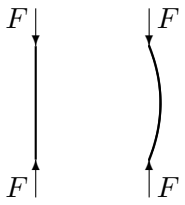

Fig. 2.6: Buckling of a rod under compression

If the compressive force F becomes large enough the sheet will buckle and deflect to the left or the right. The vertical position has gone unstable and two new symmetrical fixed points, corresponding to the left- and right-buckled configurations, have been born.

There are two very different types of pitchfork bifurcation: supercritical and subcritical pitchfork bifurcations. The first of these is simplest.

2.4.1 Supercritical pitchfork bifurcation

This has canonical form

$$\dot{x} = \mu x - x^3, \tag{2.16}$$

where μ is a real parameter. Note that this equation is *invariant* under the change of variable $x \to -x$. This is the mathematical expression of the left–right symmetry.

Fig. 2.7 shows $f(x, \mu) = \mu x - x^3$ for different values of μ, from which we can infer the fixed points.

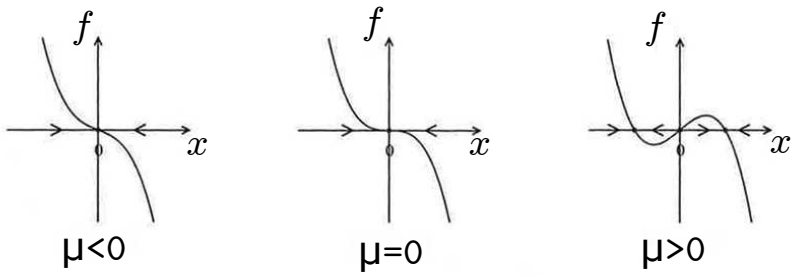

Fig. 2.7: The phase line for the supercritical pitchfork bifurcation.

Fig. 2.8 shows the bifurcation diagram, and it is clear why it is called a pitchfork bifurcation.

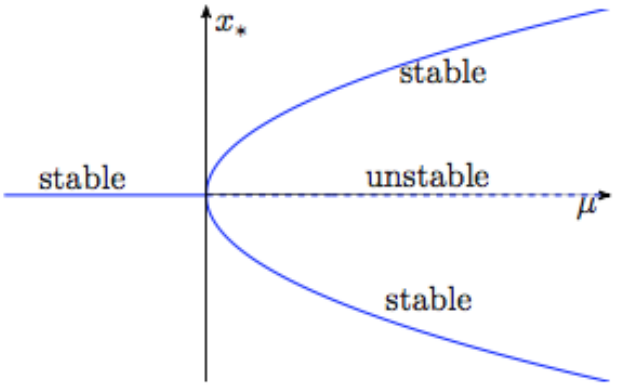

Fig. 2.8: Supercritical pitchfork bifurcation diagram.

Example Equations similar to $\dot{x} = -x + \mu \tanh x$ arise in models of magnets and neural networks. Show that this equation undergoes

a supercritical bifurcation as μ is varied and plot the bifurcation diagram shown in Fig. 2.9.

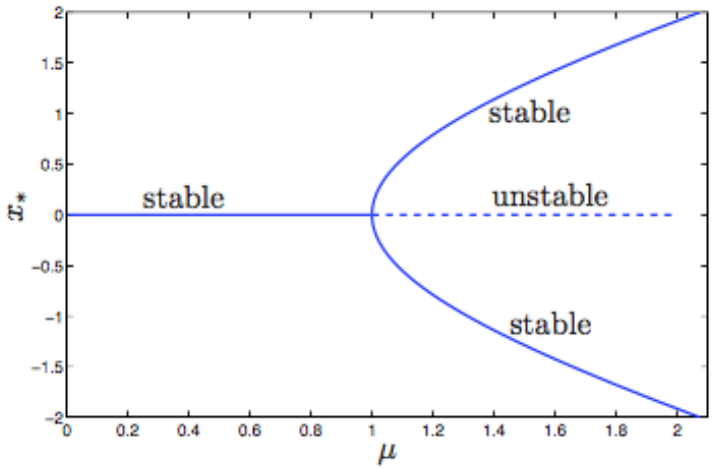

Fig. 2.9: Bifurcation diagram for $\dot{x} = -x + \mu \tanh x$.

2.4.2 Subcritical pitchfork bifurcation

In the supercritical case $\dot{x} = \mu x - x^3$, the cubic term is *stabilising*, acting as a restoring force that pulls $x(t)$ back toward $x = 0$. The subcritical pitchfork bifurcation has canonical form

$$\dot{x} = \mu x + x^3. \tag{2.17}$$

Now the cubic term is *destabilising*. From Fig. 2.10 we see that there is one unstable equilibrium $x = 0$ for $\mu \geq 0$ and three equilibria for $\mu < 0$ of which $x = 0$ is stable and the other two, $x_* = \pm\sqrt{\mu}$, are unstable.

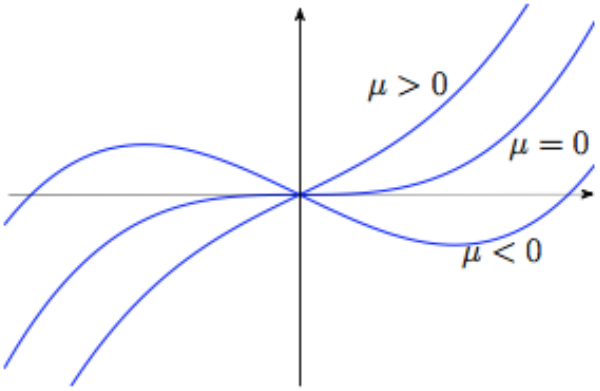

Fig. 2.10: Plot of $f(x) = \mu x + x^3$ for $\mu < 0$, $\mu = 0$ and $\mu > 0$.

The bifurcation diagram is as in Fig. 2.11; the pitchfork is inverted compared to the supercritical case. For the subcritical case, the instability for $\mu > 0$ is not opposed by the cubic term — in fact, the cubic term lends a hand to drive the trajectory $x(t)$ to infinity! This leads to blow-up; one can show that $x(t) \to \pm\infty$ in finite time, starting from any initial condition $x_0 \neq 0$.

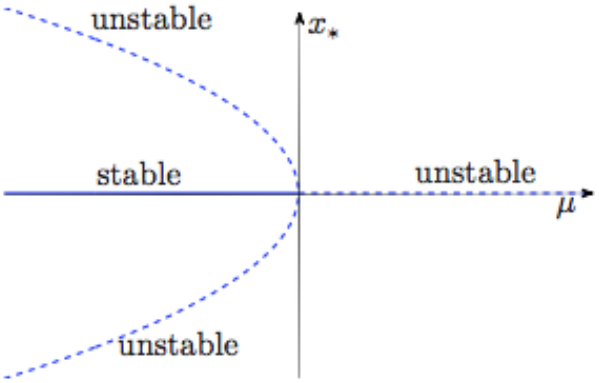

Fig. 2.11: Subcritical pitchfork bifurcation diagram.

In real physical systems, such an explosive instability is usually opposed by the stabilising influence of higher-order terms. Assuming the system is symmetric under $x \to -x$, we add the stabilising term $-x^5$:

$$\dot{x} = \mu x + x^3 - x^5. \tag{2.18}$$

The bifurcation diagram for this ODE is shown in Fig. 2.12. For small x^* this looks like Fig.e 2.11. The new feature is that the unstable branches turn around and become stable at $\mu = \mu_s$, where $\mu_s < 0$. These stable *large-amplitude* branches exist for all $\mu > \mu_s$.

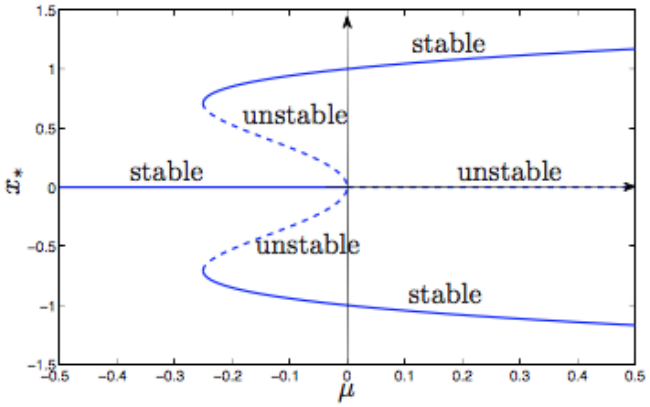

Fig. 2.12: Bifurcation diagram for $\dot{x} = \mu x + x^3 - x^5$.

How do we interpret this bifurcation diagram? We have to consider a number of cases. In the following discussion we denote the large-amplitude steady states at some value of the parameter μ by $x^* = \pm X_\mu$ and the unstable non-zero steady states by $x^* = \pm x_\mu$.

- Case 1: $\mu < \mu_s$. $\lim_{t\to\infty} x(t) = 0$ for all initial conditions $x(0) = x_0$.
- Case 2: $\mu_s < \mu < 0$. If $-x_\mu < x_0 < x_\mu$ then $\lim_{t\to\infty} x(t) = 0$. If $x_0 < -x_\mu$ then $\lim_{t\to\infty} x(t) = -X_\mu$, and if $x_0 > x_\mu$ then $\lim_{t\to\infty} x(t) = X_\mu$.
- Case 3: $\mu > 0$. If $x_0 < 0$ then $\lim_{t\to\infty} x(t) = -X_\mu$, and if $x_0 > 0$ then $\lim_{t\to\infty} x(t) = X_\mu$.

Now suppose the state of the system is $x^* = 0$ and $\mu < \mu_s$ and we then slowly increase μ. The state remains unchanged as the system travels along the stable $x^* = 0$ branch of the bifurcation curve until $\mu = 0$. When $\mu = 0^+$ the state will *jump* to one or other of the large-amplitude stable branches and then continue along that branch. Physically such a jump might be interpreted as a catastrophic event.

If we next slowly decrease the value of μ, the state will remain on the large-amplitude branch until $\mu = \mu_s$. When $\mu = \mu_s^-$ the state will again *jump*, back to the trivial steady state.

This lack of reversibility as a parameter (here μ) is varied is called *hysteresis*.

Note that the bifurcation at $\mu = \mu_s$ is a saddle-node bifurcation; stable and unstable fixed points are born as μ is increased.

2.5 The spruce-budworm model

2.5.1 The model

The spruce budworm is a significant pest species in Canada, as it can defoliate coniferous trees and devastate the forestry industry (Fig. 2.13). The first reported outbreak was in 1909 in British Columbia, and outbreaks have continued to occur in Canada and the US over increasingly large areas ever since. Some outbreaks subside naturally after a few years, but others have persisted for up to 30 years. In 1978, Ludwig *et al.* developed and studied the following model for the dynamics of the budworm population, N:

$$\frac{dN}{dt} = RN\left(1 - \frac{N}{K}\right) - p(N).$$

We assume that, in the absence of predators, the budworm population grows according to the logistic ODE, where R is the (linear) birth rate of the budworm and K is the carrying capacity (which is

Fig. 2.13: The spruce budworm.
http://en.wikipedia.org/wiki/Spruce_Budworm

related to the density of foliage available on the trees). The $p(N)$ term represents predation, e.g. by birds, and takes the form:

$$p(N) = \frac{BN^2}{A^2 + N^2}.$$

The qualitative features of this function are illustrated in Fig. 2.14.

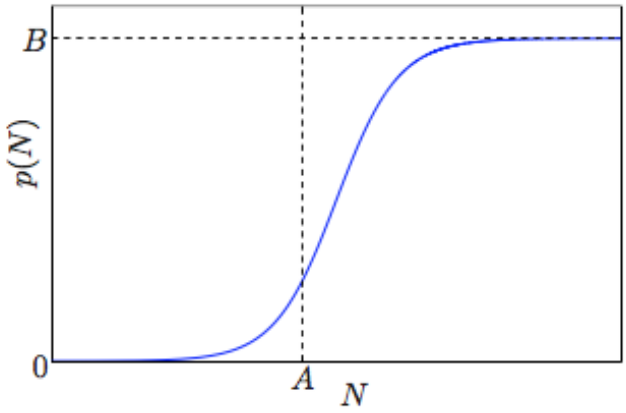

Fig. 2.14: The predation function for the spruce budworm model.

There is an approximate thresold value $N \approx A$, below which predation is low (there are few budworms, so they are difficult for predators to find, hence they concentrate on other sources of food instead). As N increases above A, the effects of predation quickly become more significant, reaching a saturation level for large N (the predators have as many budworms as they can eat so there is an upper limit on the number of budworms eaten per unit time, for a constant predator population size).

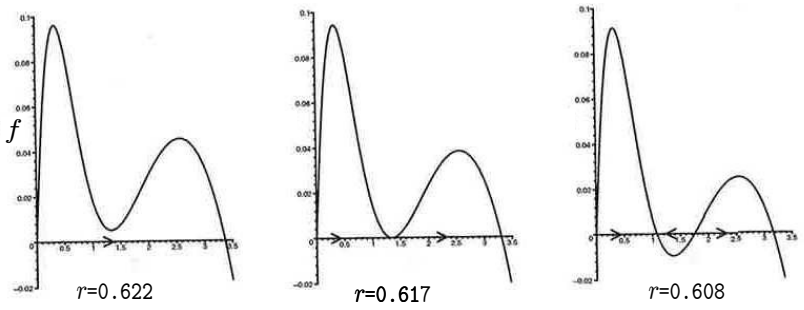

Fig. 2.15: Finding the fixed points for the spruce budworm model

The model can be written in dimensionless form (exercise) as

$$\frac{dx}{dt_*} = rx\left(1 - \frac{x}{k}\right) - \frac{x^2}{1+x^2},$$

where $k = K/A$, $r = AR/B$, $x = N/A$, $t_* = Bt/A$.

We will use this model to determine the conditions under which the budworm population will be under control and the conditions which will lead to an outbreak of budworms.

What do we mean by an "outbreak"? The idea is that as parameters drift, the budworm population suddenly jumps from a low level to a high level. $N \approx A$ or $x \approx 1$ is defined to be a low population level, so $x \gg 1$ is a high level.

2.5.2 Analysis for constant k

The right-hand-side, which we denote $f(x)$, is quite complicated as it involves two parameters, r and k. Initially, to make things simpler, we will fix $k = 6$. We plot $f(x)$ for three particular values of r in Fig. 2.15. We can see that:

1. For $r = 0.608$ there are four fixed points: $0 < x_1^* < x_2^* < x_3^*$, which are unstable, stable, unstable and stable, respectively.
2. For $r = 0.617$ there are three fixed points: $0 < x_1^* < x_2^*$ which are unstable, semi-stable and stable, respectively.
3. For $r = 0.622$ there are two fixed points: 0 (unstable) and x_1^* (stable).

We can see that as r increases from 0.608, the behaviour of the model changes fundamentally, i.e. a bifurcation occurs. What type of bifurcation is this? At first there are four fixed points. When $r = 0.617$, the curve just touches the x-axis in Fig. 2.15, and there are then only 3 fixed points; $r = 0.617$ is a bifurcation value. For $r > 0.617$, there are just two fixed points. We show the behaviour of the fixed points for a greater range of r on a bifurcation diagram.

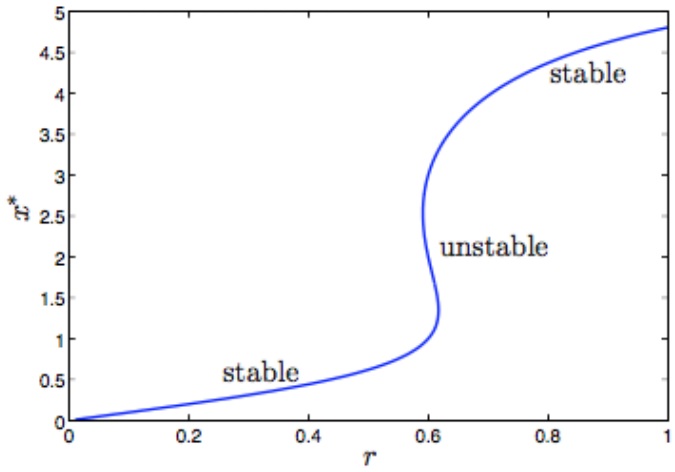

Fig. 2.16: Bifurcation diagram for the spruce budworm model, showing steady states x^* as a function of r.

Fig. 2.16 shows the bifurcation diagram. As r increases from 0 the steady state population size increases, following the lower stable branch of the bifurcation diagram. When r becomes a little larger than 0.6 the population size increases significantly — there is a jump to the upper stable branch. If r is then decreased the population size will decrease down the stable upper branch of the bifurcation diagram until this terminates and there is a jump down to the lower stable branch. We thus have a hysteresis.

Note that a change in r may be due to a change in A, R or B.

2.5.3 Analysis for two varying parameters

Now, let us write

$$f(x) = x\left[r\left(1-\frac{x}{k}\right) - \frac{x}{1+x^2}\right]. \tag{2.19}$$

Then $x = 0$ is always an unstable fixed point. It is unstable because $f(0^+) \approx rx > 0$ and $f(0^-) < 0$. Other fixed points are given by

$$r\left(1-\frac{x}{k}\right) = \frac{x}{1+x^2}, \tag{2.20}$$

an equation that is easy to analyse graphically. Fig. 2.17 graphs the left and right sides of (2.20). The number of fixed points depends on both r and k. For r sufficiently large there will be one, two or three fixed points. For small r there will be one or two fixed points.

Bifurcations occur when both (2.20) and

$$\frac{d}{dx}\left[r\left(1-\frac{x}{k}\right)\right] = \frac{d}{dx}\left(\frac{x}{1+x^2}\right) \tag{2.21}$$

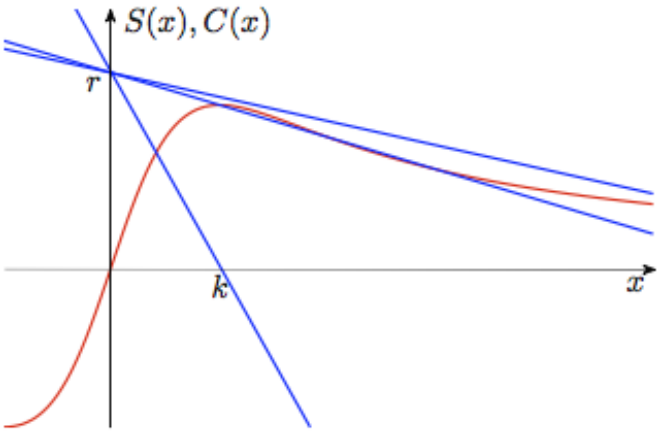

Fig. 2.17: Plots of $C(x) = x/(1+x^2)$ (red) and $S(x) = r(1-x/k)$ (blue) for different values of k. The blue curves have a y-intercept of $y = r$ and an x-intercept of $x = k$.

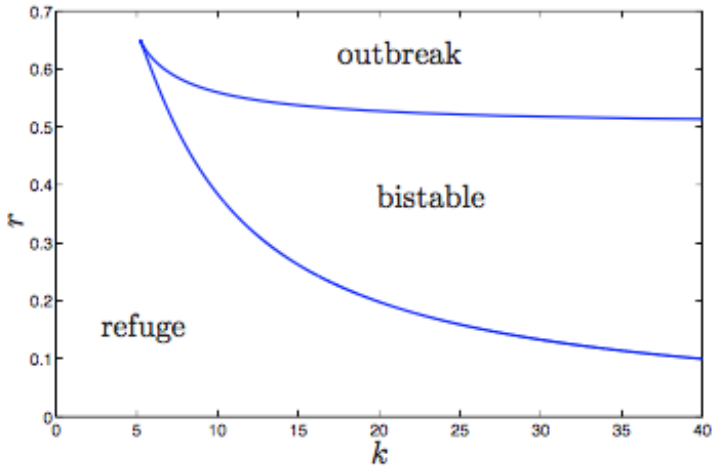

Fig. 2.18: Bifurcation curves for the spruce-budworm model.

are satisfied. From these we find

$$r = \frac{2x^3}{(1+x^2)^2}, \quad k = \frac{2x^3}{x^2-1}. \tag{2.22}$$

Since we must have $k > 0$, then $x > 1$ ($x < 0$ makes no physical sense). These equations for r and k define the bifurcation curves, i.e. curves along which bifurcations occur. For each $x > 1$ we plot the point $(k(x), r(x))$ in the (k, r) plane. The resulting curves are shown in Fig. 2.18.

For k small there is just one non-zero stable fixed point for any value r and there are no bifurcations. As k increases we enter the region where there are two bifurcations; on each of the upper and

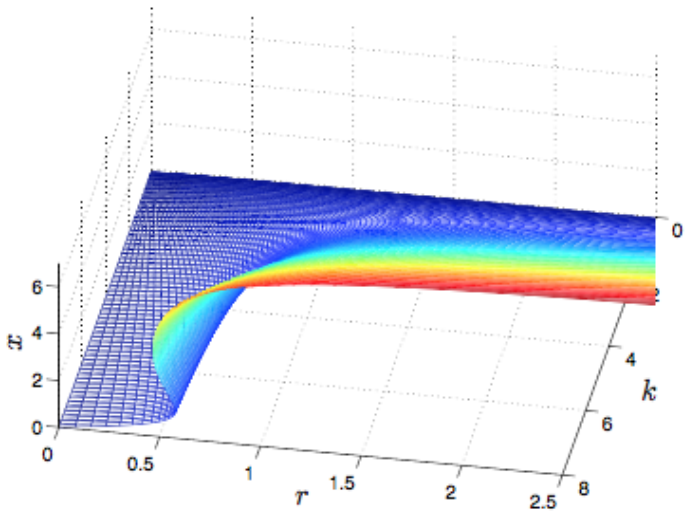

Fig. 2.19: A sketch of the bifurcation diagram for the spruce budworm model in (k, r)-space.

lower bifurcation curves there are two fixed points, between them there are three fixed points and above and below there is one.

A more complete sketch of what happens in (k, r) space is presented in Fig. 2.19.

How was this plotted?

In practical terms, the bifurcation analysis suggests that an outbreak of the pest corresponds to a small change in the environment causing a slight change in the parameter values (an increase in birth rate, say), which results in the population jumping from the lower to the upper stable branch. The outbreak subsides when the solution jumps from the upper to the lower branch. The model suggests that control of the outbreak would require us either to reduce r, or reduce k, so that the solution moves away from the bistable region in Fig. 2.18, and into the region where there is only one stable equilibrium.

See Strogatz (2000) for more discussion on the physical application of the model.

3 Solving one-dimensional ODE models

So far we have looked at

1. scaling of dimensional models to obtain dimensionless models,
2. qualitative analysis of autonomous one-dimensional ODEs using phase-line analysis.

Qualitative analysis tells us the *long-term behaviour* of a quantity, e.g. for a population, whether it will approach a steady state or become unbounded. We have also seen how a change in model parameters can affect the qualitative behaviour of the long-time solution. A sudden change in the long-time behaviour is called bifurcation and we have seen three different types of bifurcations in scalar ODEs.

Often we also want to find solutions as functions of time, to study the *transient behaviour*, as the solution evolves towards the steady state. If we are very lucky, we can solve our model exactly using methods appropriate for the ODE. But often we are not so lucky, and we must find a solution numerically. As we saw earlier, in solving an IVP, we want our problem to be well posed.

In this chapter we will:

1. review some methods for solving ODEs exactly (from DEs II or equivalent);
2. look at a theorem on existence and uniqueness of solutions;
3. consider numerical solutions of ODEs.

3.1 Exact solution methods

Here are some methods for solving first-order ODEs, that you should have seen in previous courses and should know.

3.1.1 First-order separable ODEs

$$\frac{dx}{dt} = p(t)q(x) \quad \Rightarrow \quad \int \frac{dx}{q(x)} = \int p(t)\,dt \tag{3.1}$$

Exercise In biology, assume breeding is proportional to the number n of females: $dn/dt = a(t)n$ where breeding is seasonal. Suppose $a(t) = 1 + \cos(2\pi t)$. This is separable.

Answer:

$$\int \frac{dn}{n} = \int 1 + \cos 2\pi t \, dt$$
$$\log n = t + \frac{1}{2\pi} \sin 2\pi t + A$$
$$n = B \exp\left(t + \frac{1}{2\pi} \sin 2\pi t\right)$$
If n_0 penguins at time $t = 0$
$$n_0 = B \exp\left(0 + \frac{1}{2\pi} 0\right) = B$$
$$n = n_0 \exp\left(t + \frac{1}{2\pi} \sin 2\pi t\right)$$

3.1.2 First-order linear ODEs

$$\frac{dx}{dt} + p(t)x = q(t). \tag{3.2}$$

Use the integrating factor $I(t) = \exp(\int p(t)\, dt)$ to obtain the solution

$$x(t) = \frac{C}{I(t)} + \frac{1}{I(t)} \int I(t)q(t)\, dt. \tag{3.3}$$

Or you could solve the separable homogeneous equation and find a particular solution.

Exercise Solve $du/dt + u \tan t = \sin 2t$ with $u(0) = 1$.

Answer: $I = 1/\cos t$, $\sin 2t = 2 \sin t \cos t$, and $u = 3 \cos t - 2 \cos^2 t$.

3.1.3 Exact ODEs

$$M(t, x)dt + N(t, x)dx = 0, \quad \frac{\partial M}{\partial x} = \frac{\partial N}{\partial t}. \tag{3.4}$$

Then solution is

$$F(t, x) = C, \quad \text{where} \quad \frac{\partial F}{\partial t} = M(t, x), \ \frac{\partial F}{\partial x} = N(t, x). \tag{3.5}$$

Exercise Find the general solution of $2tx^3 + 3t^2x^2 \dfrac{dx}{dt} = 0$.

Answer: $t^2x^3 = C$ or $x = kt^{-2/3}$.

3.1.4 Homogeneous ODEs

$$\frac{dx}{dt} = f\left(\frac{x}{t}\right) \tag{3.6}$$

Use the substitution $x = vt$.

Exercise Find the general solution of $\dfrac{du}{dt} = \dfrac{u^2 + 2tu}{t^2}$.

Answer: $u = \dfrac{ct^2}{1 - ct}$.

3.2 Existence and uniqueness theorem

Very often we cannot obtain an exact solution and instead seek a computational solution. In doing this we often assume that a unique (not necessarily exact) solution to a problem exists, i.e. that we have a well-posed problem. Typically this is true, but in the first lecture we saw a problem that was not well posed.

You should have already come across Picard's existence and uniqueness theorem (as it is often called) in some (perhaps simplified) form.

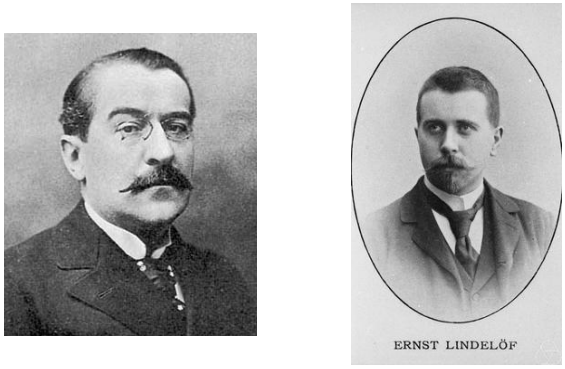

Fig. 3.1: Charles Picard (left, 1856–1941, http://en.wikipedia.org/wiki/Emile_Picard), and Ernst Lindelöf (right, 1870–1946).

Theorem 3.1. *Picard–Lindelöf theorem. Suppose $I = [t_-, t_+]$ is an interval in t with $t_- \le t_0 < t_+$ and $J = [u_-, u_+]$ is an interval in u with $u_- < u_0 < u_+$. If $f : I \times J \to \mathbb{R}$ is continuous on its domain and Lipschitz continuous (defined below) on J, then there exists $\epsilon > 0$ such that the initial value problem*

$$\frac{du}{dt} = f(t, u), \quad t \in [t_0, t_0 + \epsilon], \tag{3.7}$$

$$u(t_0) = u_0, \tag{3.8}$$

has a unique solution $u \in C^1[t_0, t_0 + \epsilon]$.

In brief, the initial value problem has a unique local solution that exists for some time interval beyond t_0. Moreover, if $t_- < t_0$ there exists $\epsilon > 0$ for which the solution exists and is unique for $t \in [t_0 - \epsilon, t_0 + \epsilon]$.

3.2.1 Lipschitz continuity

This is a measure of the smoothness of a function $f(t, u(t))$ in the vicinity of a point (t_0, u_0) where $u_0 = u(t_0)$, which is used in the Picard-Lindelöf theorem.

Definition 3.2. *Let $J = [x_0, x_1]$ be an interval on the real line. We say $f : J \to \mathbb{R}$ is **Lipschitz continuous** on J if there exists $L > 0$ such that*

$$|f(x) - f(y)| \leq L|x - y| \quad \textit{for all } x, y \in J.$$

L is a Lipschitz constant; the smallest value is the (best) Lipschitz constant.

Fig. 3.2: German-Jewish mathematician Rudolf Lipschitz (1832–1903). http://en.wikipedia.org/wiki/Rudolf_Lipschitz

---

Example Show that $f(x) = \sin(x)$ is Lipschitz continuous on $\mathbb{R}$.

In the last example, f is *globally* Lipschitz continuous, i.e. Lipschitz continuous on its entire domain. The next example is not globally Lipschitz continuous.

Example Show that $f(x) = x^2$ is Lipschitz continuous on $J = [-1, 1]$ but not on $\mathbb{R}$.

Example Show that $f(x) = |x|$ is Lipschitz continuous on $\mathbb{R}$, although it is not everywhere differentiable.

Corollary 3.3. *Let $J = [x_0, x_1]$ be an interval on the real line and $f : J \to \mathbb{R}$. Suppose $f(x)$ is continuously differentiable on J (i.e. $f \in C^1[x_0, x_1]$). Then f is Lipschitz continuous on J.*

Thus, every continuously differentiable function on J is Lipschitz continuous on J. However, a function that is Lipschitz continuous on J need not be continuously differentiable since we just need

$$\left|\frac{f(y) - f(x)}{y - x}\right| \leq L \quad \text{for all } x, y \in J \text{ with } x \neq y,$$

i.e. the magnitude of the slope of the line joining two points $(x, f(x))$, $(y, f(y))$ must be bounded. If f is differentiable we have, on taking the limit as $y \to x$, $|f'(x)| \leq L$. Lipschitz continuity is a weaker condition than continuous differentiability.

Remark 3.4. A function $f(t, x) : I \times J \to \mathbb{R}$ is Lipschitz continuous on J if there exists L such that

$$|f(t, x) - f(t, y)| \leq L|x - y| \quad \text{for all } x, y \in J \text{ and } t \in I.$$

3.2.2 Picard iteration

Picard iteration is a key tool in proving the Picard–Lindelöf theorem.

First observe that if $u(t)$ is a continuously differentiable function for $t \in [t_0, t_0 + \alpha]$, i.e. $u \in C^1[t_0, t_0 + \alpha]$, then

$$\int_{t_0}^{t} u'(s)ds = \int_{t_0}^{t} f(s, u(s))ds$$

$$\Rightarrow \qquad u(t) - u(t_0) = \int_{t_0}^{t} f(s, u(s))ds$$

and we obtain the equivalent integral form of the IVP

$$u(t) = u_0 + \int_{t_0}^{t} f(s, u(s))ds. \tag{3.9}$$

The *integral equation* form has several advantages:

- it includes the initial condition;
- it leads to easy approximate series solutions;
- it works even when there is something non-differentiable (as in the stochastic differential equations of *financial maths* where almost nothing is differentiable, but integration still works).

Exercise Write down the integral equation for the IVP

$$\frac{du}{dt} + u \tan t = \sin 2t, \; u(0) = 1.$$

To solve Eq. (3.9), let $u^{(k)}(t)$ be the kth iterate with $u^{(0)} = u_0$. Then, for $k = 1, 2, \ldots$,

$$u^{(k)}(t) = u_0 + \int_{t_0}^{t} f(s, u^{(k-1)})\, ds. \tag{3.10}$$

This is Picard iteration. It often gives a Taylor series that (hopefully) converges to the solution to the IVP, i.e.

$$\lim_{k\to\infty} u^{(k)}(t) = u(t),$$

for a large class of problems.

Example Apply Picard iteration to the IVP $u' = u,\ u(0) = 1$.

Example Apply Picard iteration to the IVP $u' = u^2,\ u(0) = 1$.

There are four steps to proving the Picard-Lindelöf theorem:

1. Show that Picard iterates (defined below) exist.
2. Show that Picard iterates converge.
3. Show that Picard iterates converge to a solution to the IVP.
4. Show that the solution is unique.

We will go through the details of these steps in lectures.

Example Establish an existence and uniqueness result for

$$\frac{du}{dt} = u, \quad u(0) = 1.$$

Example Explore existence and uniqueness for

$$\frac{dx}{dt} = x^{1/3}, \quad x(0) = 0.$$

3.3 Error, conditioning and stability

Before studying numerical solutions of first-order IVPs we discuss error and introduce some terminology. A numerical solution always contains error — we need the error to be small enough that the numerical solution is useful.

3.3.1 Error

There are two types:

1. *Truncation error* (e_T), associated with the mathematical process — typically truncation of a Taylor series.

   If a function f has $N+1$ continuous derivatives

   $$f^{(n)}(x) = \frac{d^n}{dx^n} f(x), \quad n = 1, \ldots, N+1,$$

   on the open interval (a, b) containing x and $x+h$, then

   $$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2!} f''(x) + \frac{h^3}{3!} f'''(x) + \ldots \\ &\quad + \frac{h^N}{N!} f^{(N)}(x) + R_N(x,h) \\ &= \sum_{n=0}^{N} \frac{h^n}{n!} f^{(n)}(x) + R_N(x,h), \end{aligned}$$

   where the remainder $R_N(x,h)$ is given by

   $$R_N(x,h) = \frac{h^{N+1}}{(N+1)!} f^{(N+1)}(\xi), \quad x < \xi < x+h.$$

   If we truncate the series at the $(N+1)$th term, then $e_T = R_N$ and we have an approximation of $f(x+h)$ with a truncation error of order h^{N+1}. We are assuming that the terms are becoming smaller and smaller, i.e. $h < 1$ and the derivatives of $f(x)$ are sufficiently 'nice'.

   We say that e_T is order h^m, i.e. $e_T = O(h^m)$ if there exists a constant $K > 0$ such that

   $$|e_T| \leq Kh^m.$$

   The approximation is said to be $(m-1)$th-order accurate.

   In some contexts truncation error is called *discretisation error*.

2. *Round-off or machine error* (e_R), associated with computer arithmetic. A computer can only store a finite number of digits. The error resulting from rounding or truncating of a number is called round-off error. (Machine epsilon ϵ_m is the difference between 1 and the next biggest number that a computer can store. This is much larger than the smallest positive number that can be stored exactly.)

The *total error* e is the sum of truncation and round-off error: $e = e_T + e_R$. We want to keep this small. Because we want to look at numerical solution of differential equations, we will need methods for numerically approximating derivatives. We consider finite-difference approximation of derivatives.

We can measure the *absolute* or *relative* error in an approximation. If x is the exact solution to a problem and $\bar{x}$ is an approximate solution to that problem then

$$\text{the absolute error is } e_{abs} = |x - \bar{x}|, \tag{3.11}$$

$$\text{the relative error is } e_{rel} = \left| \frac{x - \bar{x}}{x} \right|, \quad x \neq 0. \tag{3.12}$$

3.3.2 Well-conditioned and ill-conditioned problems

Recall that a problem is *well conditioned* if small changes in the data always make small changes in the solution; otherwise it is *ill conditioned*.

Consider a scalar function $f(x)$. A small change/perturbation to the input data x, say to $x + e = x(1 + \epsilon)$, $\epsilon = e/x$ and $|\epsilon| \ll 1$, yields the solution $f(x + e)$. To determine whether $f(x)$ is a well-conditioned problem we must ask whether $f(x+e) - f(x)$ is always of small magnitude. Assuming the function f to be differentiable we use a Taylor series expansion

$$f(x + e) = f(x) + e\, f'(\xi), \quad x \leq \xi \leq x + e.$$

For small e and well-behaved f, $f'(\xi) \approx f'(x)$ so that

$$f(x + e) - f(x) \approx e\, f'(x).$$

Then $f(x)$ is well conditioned if the absolute error $e\, f'(x)$ is of small magnitude for all possible input values x; otherwise it is ill conditioned.

The smallness of the error is, ideally, measured as relative error. The condition number is defined as

$$c = \left| \frac{\text{relative error in solution}}{\text{relative error in data}} \right|. \tag{3.13}$$

A well-conditioned problem has c near unity. Large c means the problem is ill-conditioned. However, for initial value problems conditioning is measured in terms of absolute error.

An IVP $dx/dt = f(t, x)$, $x(0) = C$ is ill conditioned if the curves in the family of solutions depart quickly from one another; otherwise it is well conditioned.

Example Consider the 1st-order IVP

$$\frac{dx}{dt} = -x, \quad x(0) = 1.$$

Example Now consider the problem

$$\frac{dx}{dt} = x, \quad x(0) = 1.$$

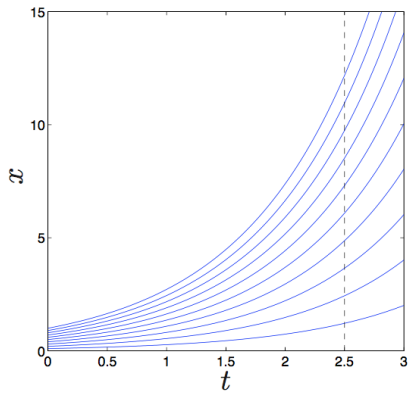

Fig. 3.3: Solution curves of $dx/dt = x$, $x(0) = C$ for $C = 0.1, 0.2, \ldots, 0.5$. A small change in the initial condition results in a large change in $x(t)$ at large t.

More precisely, for a first-order IVP $dx/dt = f(t,x)$, $x(t_0) = c$:

- If $\partial f/\partial x < 0$ the problem is well conditioned (curves approach one another as x increases).
- If $\partial f/\partial x > 0$ the problem is ill conditioned (curves depart from one another as x increases).
- If $\partial f/\partial x = 0$ the problem is on the boundary between well and ill conditioned.

3.3.3 Stable and unstable algorithms

An algorithm is *(numerically) stable* if it always produces the solution to a nearby problem; otherwise it is *(numerically) unstable*.

Again consider a scalar function $f(x)$ and suppose that we evaluate it using an algorithm $\text{alg}(x)$. The absolute error in the solution is $\text{alg}(x) - f(x)$. Suppose that we can show that $\text{alg}(x) = f(x+e)$ for some "small" e, i.e. the algorithm yields the solution to the nearby problem of evaluating f at $x+e$. Then the algorithm is *numerically stable*, or just *stable* (for short).

When solving an IVP, the error (truncation and/or round-off) introduced at one step affects the accuracy of the solution at the next and subsequent steps — the error *propagates* (see Fig. 3.4). One can compare this to an interest process; in each stage there is "interest" on previously committed errors at the same time as a new "error capital" is put in. The interest rate can, however, be negative, an advantage in this context. If an error grows in magnitude as it propagates, the solution method is unstable and we will get the solution to a problem that is very far from that we were trying

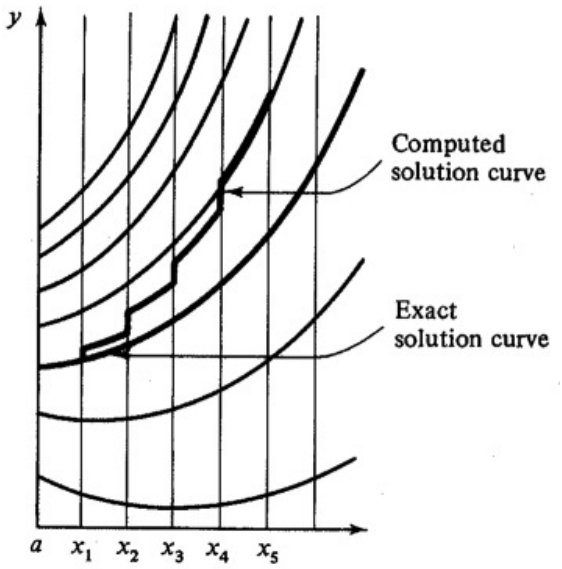

Fig. 3.4: Figure 8.1.4 from Dahlquist & Björck (2003).

to solve; if it decays as it propagates, then the solution method is stable.

The study of error propagation is called error analysis or stability analysis.

3.3.4 Conditioning and stability

It is important to distinguish between difficult problems and bad algorithms. The numerical solution to a problem may be bad because:

1. the problem is ill-conditioned;
2. the numerical method used is not stable;
3. or both of these.

To get a good solution we need a well-conditioned problem and a stable numerical method — the solution will be to a problem near to the one we are trying to solve (stability) and the error in the solution will be small (well conditioned).

If we have a well-conditioned problem and use an unstable algorithm, then the solution will be that of a problem very far from the problem we are trying to solve (unstable) and we cannot expect a solution that is close to accurate. But we can rectify this by using a stable algorithm.

If we have an ill-conditioned problem we will get a bad solution. Even if we use a stable algorithm, so that the solution is of a problem close to that we are trying to solve, the solution will contain large

error because the problem is ill conditioned. Sometimes we can reformulate the problem to render it better conditioned, but this is not always possible. It is difficult to get accurate numerical solutions to ill-conditioned problems.

As we consider numerical solution of IVPs we will assume that our problems are well conditioned and focus on numerical stability.

3.4 Numerical solution of 1st-order initial value problems

To solve the IVP

$$\frac{dx}{dt} = f(x,t), \quad x(t_0) = x_0, \tag{3.14}$$

we need to approximate the first derivative. This is readily done using Taylor series.

3.4.1 Numerical schemes

From Taylor's expansion we have

$$x(t+h) = x(t) + hx'(t) + \frac{h^2}{2}x''(\xi).$$

Rearranging gives

$$x'(t) = \frac{x(t+h) - x(t)}{h} - \frac{h}{2}x''(\xi).$$

Neglecting the truncation error $e_T = \mathrm{O}(h) \to 0$ as $h \to 0$ we have a finite difference approximation to $x'(t)$.

Commonly we have data t_j, $x_j = f(t_j)$ at equispaced points $t_j = t_0 + jh$. Then,

$$\begin{aligned} x'_j = x'(t_j) &= \frac{x(t_j+h) - x(t_j)}{h} + \mathrm{O}(h) \\ &= \frac{x_{j+1} - x_j}{h} + \mathrm{O}(h). \end{aligned}$$

Dropping the $O(h)$ terms gives the 1st-order *forward* difference formula for $x'(t)$, $x'_j = (x_{j+1} - x_j)/h$ which is first-order accurate, i.e. $|e_T| \leq Kh$ for some constant K.

This yields Euler's method, or the forward Euler method, for the IVP:

$$\frac{x_{j+1} - x_j}{h} + O(h) = f(x_j, t_j) \;\Rightarrow\; x_{j+1} = x_j + hf(x_j, t_j) \tag{3.15}$$

which has *local discretisation error* $\ell(h) = O(h^2)$, which is the truncation error in the derivative approximation multiplied by h. It is local because it is the discretisation error in one step.

Usually we want to compute out to some time $t_N = t_0 + Nh$ starting from t_0 where we know that $x(t_0) = x_0$, where N is the number of steps taken The *global discretisation error* is the total discretisation error over the N steps:

$$g(h) = N \times O(h^2) = \frac{t_N - t_0}{h} \times O(h^2) = O(h).$$

We say that Euler's method is first-order accurate.

Example Obtain the 1st-order backward difference formula for $x'(t)$ using the Taylor series for $x(t-h)$. Demonstrate that it has a global discretisation error $O(h)$.

The backward difference formula for the first derivative yields the backward Euler method for the IVP:

$$x_j = x_{j-1} + hf(x_j, t_j). \tag{3.16}$$

A more accurate, 2nd-order finite difference formula for $x'(t)$ results in

$$x_{j+1} = x_{j-1} + 2hf(x_j, t_j). \tag{3.17}$$

This is known as the *explicit midpoint method* or the *leap-frog method*.

The forward and backward Euler methods are commonly used for first-order scalar IVPs. Other numerical schemes may be obtained using other approximations to the first derivative. For example, they can be derived using the method given below for approximating the rth derivative of a function $x(t)$, $x^{(r)}$. Typically, the more data points used in the approximation the better the accuracy — which, actually, does not mean a better numerical scheme! The leap-frog method, though having a higher order of accuracy than the Euler methods, is not a generally acceptable method, as you will see.

Finite difference approximations to $x^{(r)}$

In general, assuming that x is differentiable and continuous as far as required, coefficients a_i may be found to satisfy

$$x^{(r)}(t_j) = x_j^{(r)} = \sum_{i=-p}^{q} a_i x_{j+i} + \mathrm{O}(h^m), \quad h, p, q, m > 0$$

by expanding each x_{j+i} as a Taylor series and solving for the a_i. For the rth derivative we have to eliminate all lower (1st to $(r-1)$th) derivatives and x_j, and solve for the coefficient of $x_j^{(r)}$ (=1), so we obtain $r+1$ equations in terms of $p+q+1$ unknowns. Then, to solve we need $p + q \geq r$. If $p + q > r$ we can eliminate higher

order terms in the Taylor expansion and obtain a more accurate difference formula.

Let's illustrate by deriving the difference formula for $x'(t)$ with $p = 0, q = 2$:

$$\begin{aligned} x_j^{(1)} &= \sum_{i=0}^{2} a_i x_{j+i} + \mathrm{O}(h^m) \\ &= a_0 x_j + a_1 x_{j+1} + a_2 x_{j+2} \mathrm{O}(h^m) \\ &= a_0 x_j + a_1 \left(x_j + h x_j' + \frac{h^2}{2!} x_j'' \right. \\ &\qquad \left. + \frac{h^3}{3!} x_j''' + \frac{h^4}{4!} x_j^{(4)} + \dots \right) \\ &\quad + a_2 \left(x_j + (2h) x_j' + \frac{(2h)^2}{2!} x_j'' \right. \\ &\qquad \left. + \frac{(2h)^3}{3!} x_j''' + \frac{(2h)^4}{4!} x_j^{(4)} + \dots \right). \end{aligned}$$

Then

$$\begin{aligned} a_0 + a_1 + a_2 &= 0, \quad \text{eliminates } x_j \text{ terms} \\ h(a_1 + 2a_2) &= 1, \\ a_1 + 4a_2 &= 0, \quad \text{eliminates } x_j'' \text{ terms,} \end{aligned}$$

which yields $a_1 = 4/(2h)$, $a_2 = -1/(2h)$ $a_0 = -3/(2h)$, and hence

$$x_j' = \frac{-3x_j + 4x_{j+1} - x_{j+2}}{2h}.$$

To find the order of the error we need to look at the x_j''' terms. We have

$$\frac{h^3}{3!}(a_1 + 8a_2) = -\frac{h^3}{3!}\frac{2}{h} = O(h^2),$$

so that $m = 2$ and the formula is 2nd-order accurate. Note that you must check that the coefficients don't sum to zero; sometimes they do and the method is more accurate than a cursory glance will suggest.

Clearly the more terms are retained in the Taylor series, the more data points are used to approximate the derivative, and the better the accuracy of the difference formula.

A Simple Check

Observe that in all the finite-difference formulas, the sum of all the coefficients of the function values (f_j) appearing in the numerator is zero. This is always so and physically implies that the derivative becomes zero if $f(x)$ is a constant function.

3.4.2 Explicit and implicit methods

We have derived three methods for solving the IVP

$$\frac{dx}{dt} = f(t,x), \quad x(0) = a,$$

for $x(t),\ t > 0$:

1. The forward Euler method, or just Euler's method, also called the explicit Euler method, $x_{i+1} = x_i + hf_i$.
2. The backward Euler method, also called the implicit Euler method, $x_i - hf_i = x_{i-1}$, $x_0 = a$.
3. The leapfrog or explicit midpoint method, $x_{i+1} = x_{i-1} + 2hf_i$.

For each of these methods we have the starting point $x_0 = a$.

The forward Euler method is an *explicit method* — we can determine x at t_{i+1} using only information from the previous point t_i. Given $x_0 = a$ we can compute $x_1, x_2, \ldots$. As is typical with all numerical solution methods, we obtain the values of the function $x(t)$ at a set of discrete points $t_1, t_2, \ldots$.

The backward Euler method is an *implicit method* — we have the job of determining x at t_i using information from the previous point and the current one. Depending on the function f this may or may not be hard.

Example For the ODE $dx/dt = x$ the explicit Euler method gives $x_{i+1} = x_i(1+h)$ and the implicit method gives $x_i = x_{i-1}/(1-h)$, both readily evaluated.

Example What are the explicit and implicit Euler formulas for $dx/dt = e^{-x}$? Are these readily evaluated?

Exercise Is the leapfrog method an explicit or implicit method? Can you see a problem with this method? What can be done about it?

3.4.3 Consistency

A finite-difference formula is *consistent* with an ODE if the local discretization error $\ell(h) \to 0$ as $h \to 0$. This is equivalent to saying that, at any fixed point t_i in the domain, the finite difference formula becomes the same as the ODE as $h \to 0$. We can do a *consistency analysis* to find out.

Fig. 3.5: American–Jewish mathematician Peter Lax (born 1926). Born in Hungary and emigrated with his parents in 1941 to escape Nazism. http://en.wikipedia.org/wiki/Peter_Lax

Consistency analysis for Euler's method

$$\begin{aligned} x_{i+1} &= x_i + hf(x_i, t_i), \\ x(t_i + h) &= x(t_i) + hx'(t_i) + \frac{h^2}{2}x''(\xi) = x(t_i) + hf(x_i, t_i), \\ x'(t_i) &= f(x_i, t_i) - \frac{h}{2}x''(\xi). \end{aligned}$$

As $h \to 0$ we have the ODE $x'(t_i) = f(x_i, t_i)$, so Euler's method is consistent with the first-order ODE $x' = f(x, t)$.

Note that a consistency analysis gives us the local discretisation error and the truncation error resulting from the finite-difference approximation to the derivative. From the above we have

$$\begin{aligned} & \frac{x_{i+1} - x_i}{h} = x'(t_i) + \frac{h}{2}x''(\xi) \\ \Rightarrow \quad & e_T = \frac{x_{i+1} - x_i}{h} - x'(t_i) = \frac{h}{2}x''(\xi). \end{aligned}$$

3.4.4 Convergence

We have *convergence* to the true solution if the global discretization error $g(h) \to 0$ as $h \to 0$. Convergence is a stronger requirement than consistency and is essential for an accurate solution.

Lax's Equivalence Theorem

$$\text{Consistency} + \text{stability} = \text{convergence}.$$

3.4.5 Stability

Recall, an algorithm is unstable if the solution obtained is for a problem that is far from the problem we intended to solve.

Example Let's apply Euler's method to the well-conditioned problem

$$\frac{dx}{dt} = -100x, \quad x(0) = 1,$$

for $t > 0$ (analytic solution $x = e^{-100t}$).

For this problem, Euler's method is *conditionally stable* — stable only for certain values of h.

In comparison, the backward-Euler method is *unconditionally stable* for this problem.

Examination of the explicit and implicit Euler methods

We can better understand what is happening here by looking at things more generally. Hence we consider the ODE

$$\frac{dx}{dt} = f(t, x).$$

Let $x(t_n)$ be the exact solution at $t_n = nh$, and x_n be the approximate numerical solution. First consider Euler's method,

$$x_{n+1} = x_n + hf(t_n, x_n).$$

Taylor's series,

$$x(t + h) = x(t) + hx'(t) + \frac{h^2}{2}x''(\xi),$$

tells us that if x_n is correct, i.e. $x_n = x(t_n)$, then

$$x(t_{n+1}) - x_{n+1} = \frac{h^2}{2}x''(\xi), \quad \text{the local discretization error,}$$

where we are neglecting roundoff error. But as we iterate with Euler's method, x_n has some error, so that

$$x(t_{n+1}) - x_{n+1} = x(t_n) - x_n + h(f(t_n, x(t_n)) - f(t_n, x_n)) + \frac{h^2}{2}x''(\xi).$$

Using the mean value theorem we may write

$$f(t_n, x(t_n)) - f(t_n, x_n) = (x(t_n) - x_n)f_x(t_n, \eta),$$

where $\min\{x(t_n), x_n\} \le \eta \le \max\{x(t_n), x_n\}$, so that

$$x(t_{n+1}) - x_{n+1} = (x(t_n) - x_n)(1 + hf_x(t_n, \eta)) + \frac{h^2}{2}x''(\xi).$$

Now $x(t_n) - x_n$ is the global error, assuming no roundoff error, at t_n. Hence we have

$$\begin{aligned}(\text{global error})_{n+1} =& (1 + hf_x(t_n, \eta))(\text{global error})_n \\ &+ (\text{local error})_{n+1},\end{aligned}$$

and we can see that global errors are magnified, and Euler's method is unstable, if $|1 + hf_x(t_n, \eta)| > 1$. Then Euler's method is stable for

$$|1 + hf_x(t_n, \eta)| < 1 \quad \Rightarrow \quad -2 < hf_x(t, x) < 0.$$

Exercise Work through the backward Euler method in the same way and find its stability interval.

You should find that the interval of stability for the backward Euler method is

$$|1 - hf_x(t_n, \eta)| > 1, \quad \Rightarrow \quad hf_x(t, x) < 0, \quad hf_x(t, x) > 2.$$

For $dx/dt = -100x$, $x(0) = 1$, $t > 0$, we have $f_x = -100$. Then

- $hf_x < 0$ for all h so that the backward Euler method is *unconditionally stable*. However, for the forward Euler method we require $-2 < -100h < 0$ for stability or $0 < h < 0.02$, and the method is *conditionally stable*.
- For $dx/dt = 100x$, Euler's method is *unconditionally unstable* and the backward Euler method is *conditionally stable* ($h > 0.02$). Interesting: too small a time step is a problem! Of course, too large a time step will mean a large local error.

The stability region of a numerical method

Very often we use *test problems* to get insight into how a numerical method functions. These test problems are simple enough to be analyzed theoretically but still so general that they can present some difficulty for a prospective numerical method (Dahlquist & Björck 2003).

For numerical treatment of ODEs, the simple test problem generally used is

$$\frac{dy}{dx} = \lambda y, \quad y(0) = 1, \tag{3.18}$$

where λ is a (complex) constant.

For most numerical methods, the long-range behaviour of the solutions of the test problem depends on the quantity $f_y h = \lambda h$, where h is the grid spacing.

The region of stability of a numerical method for an initial-value problem is that set of (complex) values of $z = \lambda h$ for which all solutions of the test problem (3.18) will remain bounded as $n \to \infty$.

Thus, for the explicit Euler method the region of stability is

$$|1 + \lambda h| < 1,$$

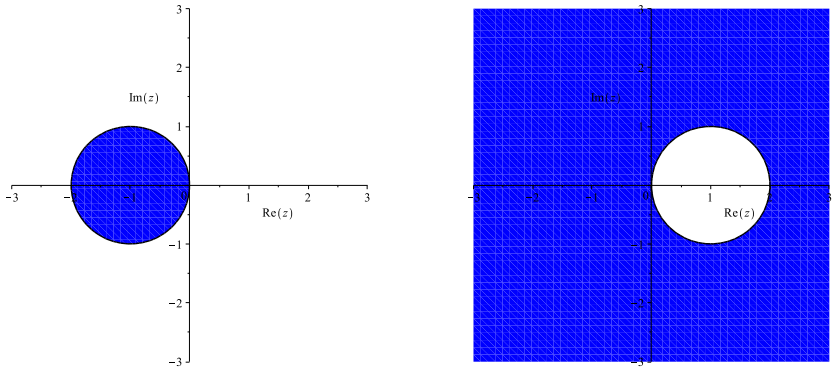

Fig. 3.6: Stability diagrams for the (left) explicit and (right) implicit Euler methods.

the interior of the unit circle in the complex plane with centre $z = -1$. For the implicit Euler method the region of stability is

$$|1 - \lambda h| > 1,$$

the exterior of the unit circle in the complex plane with centre $z = 1$.

It is desirable to guarantee that the numerical approximation to a problem is bounded when the exact solution is bounded. This means we are especially interested in numerical methods for which the stability region includes all of the left half plane (i.e. $\lambda h < 0$). Methods with this desirable property are said to be 'A-stable'. The implicit Euler method is A-stable; the explicit Euler method is not.

Plots of stability regions for a variety of methods are given in *Numerical Methods for Ordinary Differential Equations*, Butcher (2003).

Example Stability region for Heun's method

Heun's method, or the modified Euler method, is

$$\begin{aligned} x^*_{n+1} &= x_n + hf(t_n, x_n), \\ x_{n+1} &= x_n + \frac{h}{2}\left(f(t_n, x_n) + f(t_{n+1}, x^*_{n+1})\right). \end{aligned}$$

Find the stability region for this method.

3.4.6 Can we reduce the step size as much as we please?

From what we have done to date it would appear that reducing the step size h will always give a more accurate solution. In fact this is not entirely true. The reason for this is round-off error.

Consider the forward difference formula for the 1st derivative

$$x'_j = \frac{x_{j+1} - x_j}{h}$$

which has truncation error $e_T = -\frac{h}{2}x''(\xi)$, $t_j < \xi < t_{j+1}$. Besides the truncation error, there is round-off error (or, perhaps, measurement error) in the given data. Say

$$\epsilon_j = x(t_j) - x_j, \quad \text{or} \quad x_j = x(t_j) - \epsilon_j,$$

where $x(t_j)$ is the true value of x at $t = t_j$ and x_j is the given value. Then

$$\begin{aligned} e =& x'(t_j) - \frac{x_{j+1} - x_j}{h} \\ =& x'(t_j) - \frac{[x(t_{j+1}) - \epsilon_{j+1}] - [x(t_j) - \epsilon_j]}{h} \\ =& \left[x'(t_j) - \frac{x(t_{j+1}) - x(t_j)}{h}\right] + \frac{\epsilon_{j+1} - \epsilon_j}{h} \\ =& e_T + e_R. \end{aligned}$$

Now, $|e| = |e_T + e_R| \leq |e_T| + |e_R|$ where

$$\begin{aligned} |e_T| =& \frac{h}{2}|x''(\xi)| \leq \frac{h}{2}\left[\max_{t_j<t<t_{j+1}} |x''(t)|\right] = \frac{Kh}{2}, \\ |e_R| =& \frac{|\epsilon_{j+1} - \epsilon_j|}{h} \leq \frac{|\epsilon_{j+1}| + |\epsilon_j|}{h} \leq \frac{2\epsilon}{h}, \end{aligned}$$

ϵ being the upper bound on error in given data. So

$$|e| \leq M = \frac{Kh}{2} + \frac{2\epsilon}{h}.$$

As $h \to 0$ the truncation error decreases, but the round-off error increases and the value of h that minimises the total error is such that

$$\frac{K}{2} - \frac{2\epsilon}{h^2} = 0 \quad \Rightarrow \quad h_m = 2\sqrt{\frac{\epsilon}{K}}.$$

See Figure 3.7.

3.4.7 Stiff Problems

Consider

$$\frac{dx}{dt} = -\alpha x, \quad x(0) = 1, \quad \text{where } \alpha > 0.$$

This has a smooth solution $x = e^{-\alpha t}$ and $x \to 0$ as $t \to \infty$. Initially x changes rapidly with t — the larger α, the faster; then the solution changes slowly and becomes essentially constant, $x = 0$ (sketch graph).

We would expect to use a small step size where the solution is changing rapidly and be able to use a larger step size where the solution is changing slowly.

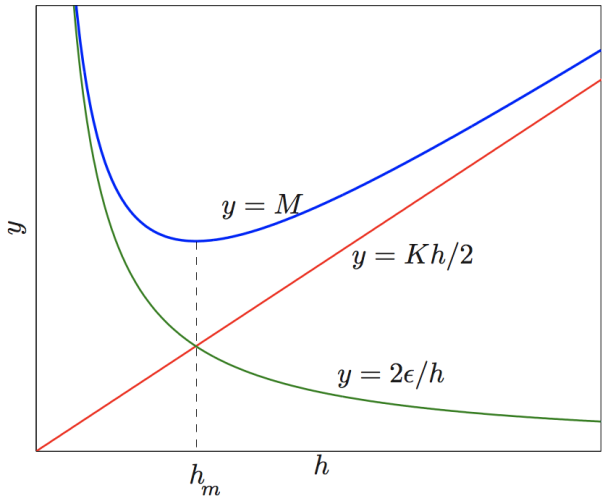

Fig. 3.7: Truncation, round-off and total error.

Suppose we use Euler's method $x_{n+1} = (1-\alpha h)x_n = (1-\alpha h)^{n+1}x_0$. This is stable (global error is not magnified) for

$$|1 - \alpha h| < 1$$

and, hence, $0 < \alpha h < 2$ or $0 < h < 2/\alpha$. The step size must be inversely proportional to the value of α; the larger the value of α the smaller must be the step size h, even where x changes slowly.

There is a "boundary layer" near $t = 0$ of size $\sim 1/\alpha$ (by the time $t = 1/\alpha$ the solution is $x = e^{-1}$). But, if we compute well beyond this time, the actual time scale of interest is much larger than this — the problem then contains two significantly different time scales.

In general a problem, like this, is called *stiff* if the actual time scale is (much) larger than small time scales potentially existing in the problem. The phenomenon that accuracy requirements following from smoothness of the solutions allow for larger time steps, whereas stability arguments, such as in the forward Euler method, require smaller time steps, relates to the stiffness of the problem (Mattheij et al. 2005, p. 95).

In this sense the problem is not stiff if our interest is confined to the boundary layer.

Of course, we know that the implicit (backward Euler) method is unconditionally stable for this problem, so with this method a large step size will be fine in the region where the solution changes slowly. The problem is not stiff in relation to this method.

3.4.8 Numerical methods for solving IVPs

The purpose of this section is to give you understanding of why different formulas are needed.

Table 3.1: Numerical methods for solving first-order IVPs using equal stepsizes h.

Some Adams-Bashforth formulas	Order	Local error
$y_{n+1} = y_n + hf_n$	$k=1$	$\frac{h^2}{2}y''(\xi)$
$y_{n+1} = y_n + \frac{h}{2}(3f_n - f_{n-1})$	$k=2$	$\frac{5h^3}{12}y'''(\xi)$
$y_{n+1} = y_n + \frac{h}{12}(23f_n$ $-16f_{n-1} + 5f_{n-2})$	$k=3$	$\frac{3h^4}{8}y''''(\xi)$

Some Adams-Moulton formulas	Order	Local error
$y_{n+1} = y_n + hf_{n+1}$	$k=1$	$-\frac{h^2}{2}y''(\xi)$
$y_{n+1} = y_n + \frac{h}{2}(f_n + f_{n+1})$	$k=2$	$-\frac{h^3}{12}y'''(\xi)$
$y_{n+1} = y_n + \frac{h}{12}(5f_{n+1}$ $+8f_n - f_{n-1})$	$k=3$	$-\frac{h^2}{2}y''(\xi)$

The explicit 4th-order Runge-Kutta method		
$y_{n+1} = y_n + \frac{1}{6}(k_1$ $+2k_2 + 2k_3 + k_4)$	$k=4$	$O(h^5)$
$k_1 = hf(x_n, y_n),$		
$k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}),$		
$k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}),$		
$k_4 = hf(x_n + h, y_n + k_3)$		

The explicit Euler method is simple to use, but if a small step size is needed for stability it can be computationally expensive and another method might be better.

You might also choose a different method if you want better accuracy (e.g. Euler methods give only first-order accuracy).

Numerical methods books will, typically, give a variety of different methods that can be used, along with their accuracy and, hopefully, their regions of stability. Some different "Adams formulas" are given in Table 3.1. These have the general form

$$y_n = y_{n-1} + h(\beta_0 f_n + \beta_1 f_{n-1} + \beta_2 f_{n-2} + \cdots + \beta_k f_{n-k}),$$

where the β_i are chosen to give the highest possible order. For an explicit method we set $\beta_0 = 0$, else the method is implicit. Adams methods are the most important linear multistep methods for nonstiff problems (Butcher 2003, p.98). Multistep methods use the solution at multiple previous steps to compute the solution at the current step (you need a one-step method to get started); the

explicit and implicit Euler methods are one-step Adams methods. Explicit Adams formulas are called "Adams-Bashforth formulas", implicit formulas are called "Adams-Moulton formulas".

Alternatives to Adams formulas include Runge-Kutta formulas. The very popular 4th-order Runge-Kutta formula is given in Table 3.1.

Special methods are used for stiff problems, e.g. implicit Runge-Kutta methods (Butcher 2003, p. 86).

Matlab ODE solvers

Matlab has a variety of different ODE solvers. There are some specifically for stiff problems. You will explore the Matlab solvers a little in your assignment/tutorial work.

3.4.9 Nonlinear first-order initial value problems

So far we have only considered numerical solution of linear IVPs. Nonlinear IVPs arise in many applications and, most often, don't have exact solutions. How are these solved?

Nonlinear problems are as easily solved using explicit methods as are linear problems. For example

$$\frac{dx}{dt} = -x^2, \quad x(0) = 1,$$

can be solved using the explicit Euler method:

$$x_{n+1} = x_n - hx_n^2,$$

which is stable for $|1-2hx| < 1$, i.e. $0 < h < 1$ since x is decreasing. Clearly, for some RHS functions you need to be careful about the choice of step size to ensure stability throughout the computation.

They are not so easily solved using implicit methods. To use these we can *linearise* the ODE:

$$\frac{dx_n}{dt} \approx -x_n x_{n-1}.$$

The implicit Euler method is then

$$x_n = x_{n-1} - hx_n x_{n-1} \quad \text{or} \quad x_n = \frac{x_{n-1}}{1 + hx_{n-1}},$$

which is unconditionally stable for $x > 0$.

More usually we use predictor-corrector (PECE) method.

3.4.10 Predictor-corrector methods

These methods can be used for both linear and nonlinear problems. The predictor-corrector algorithm is as follows:

For $n = 0, \ldots, N-1$:

P (predict): Estimate x_{n+1} using an explicit-method formula, to give x^P_{n+1},

E (evaluate): Evaluate $f_{n+1} = f(t_{n+1}, x^P_{n+1})$ using the estimate x^P_{n+1},

C (correct): Re-estimate x_{n+1} using an implicit-method formula,

E (evaluate): Evaluate f_{n+1} using improved information.

Using the explicit and implicit Euler methods for the IVP

$$dx/dt = f(t,x),\ x(a) = c,$$

gives:

$$\begin{array}{lrcl} \text{P:} & x^P_{n+1} & = & x_n + hf(t_n, x_n) \\ \text{E:} & f^P_{n+1} & = & f(t_{n+1}, x^P_{n+1}) \\ \text{C:} & x^C_{n+1} & = & x_n + hf^P_{n+1} \\ \text{E:} & f^C_{n+1} & = & f(t_{n+1}, x^C_{n+1}) \end{array}$$

If the difference between x^P_{n+1} and x^C_{n+1} is too large we can repeat the CE steps, although reducing the step size is generally a safer cure. If repeated m times it is a PE(CE)^m scheme.

An estimate of the local error in this method is as follows:

$$\begin{aligned} x^P_{n+1} - x(t_{n+1}) &= \frac{h^2}{2}x''(\xi_1), \\ x^C_{n+1} - x(t_{n+1}) &= -\frac{h^2}{2}x''(\xi_2), \end{aligned}$$

so that

$$x^P_{n+1} - x^C_{n+1} = \frac{h^2}{2}(x''(\xi_1) + x''(\xi_2)) \approx \frac{2h^2}{2}x''(\xi),$$

for $t_n \le \xi \le t_{n+1}$, which is readily computable. If $|x^P_{n+1} - x^C_{n+1}|$ is large we should worry about local error. Practical methods for solving ODEs use such estimates for the local error (even for linear problems) to determine whether the current choice of stepsize h is adequate.

3.5 Numerical solution of higher order IVPs

These can be reduced to a system of first-order IVPs, i.e. a first-order *vector* IVP. Because we will, in this course, soon be interested in models comprised of systems of ODEs, we will consider numerical solution of these.

We will also look at an alternative method for second-order IVPs, namely applying finite difference methods directly in a similar manner to first-order IVPs.

3.5.1 First-order vector IVPs

Consider an nth order ODE

$$\frac{d^n x}{dt^n} = f(t, x, x', x'', , \ldots, x^{(n-1)})$$

with initial conditions

$$x(0) = c_0,\ x'(0) = c_1,\ \ldots,\ x^{(n-1)}(0) = c_{n-1}.$$

Let $y_0 = x$, $y_1 = x'$, $\ldots y_{n-1} = x^{(n-1)}$. Then the IVP may be written

$$\begin{aligned} y_0' &= y_1, \quad y_0(0) = c_0, \\ y_1' &= y_2, \quad y_1(0) = c_1, \\ y_2' &= y_3, \quad y_2(0) = c_2, \\ &\vdots \\ y_{n-1}' &= f(t, y_0, y_1, \ldots, y_{n-1}), \quad y_{n-1}(0) = c_{n-1}. \end{aligned}$$

If $f(t, y_0, \ldots, y_{n-1})$ is linear in $y_0, \ldots, y_{n-1}$ then we may write this in matrix form:

$$\mathbf{y}' = M\mathbf{y}, \quad \mathbf{y}(0) = \mathbf{c}, \tag{3.19}$$

where $\mathbf{y} = (y_0, y_1, \ldots, y_{n-1})^T$ and

$$M = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & 0 & & 1 \\ \alpha_0(t) & \alpha_1(t) & \alpha_2(t) & \ldots & \alpha_{n-1}(t) \end{bmatrix}$$

for some functions $\alpha_j(t)$, $j = 0, 1, \ldots, n-1$.

Example Write $x'' = x,\ x(0) = 1,\ x'(0) = -1$ as a vector IVP.

It's not too hard to solve such a system of equations using the explicit Euler method — exercise. Let's look at solving using the implicit Euler method. We obtain after some manipulation

$$\begin{aligned} y_0^m - h y_1^m &= y_0^{m-1}, && (3.20) \\ y_1^m - h y_2^m &= y_1^{m-1}, && (3.21) \\ y_2^m - h y_3^m &= y_2^{m-1}, && (3.22) \\ &\vdots \\ y_{n-1}^m - h f(t, y_0^m, y_1^m, \ldots, y_{n-1}^m) &= y_{n-1}^{m-1}. && (3.23) \end{aligned}$$

Assuming linear ODEs, these equations can be written as a matrix system and solved simultaneously:

$$M\mathbf{y}^m = \mathbf{y}^{m-1}, \quad \mathbf{y}^0 = \mathbf{c},$$

where

$$M = \begin{bmatrix} 1 & -h & & & \\ & 1 & -h & & 0 \\ 0 & & \ddots & \ddots & \\ & & & 1 & -h \\ \beta_0 & \beta_1 & \beta_2 & \cdots & 1+\beta_{n-1} \end{bmatrix}$$

$\beta_j = -h\alpha_j(t_m)$. If the ODE is nonlinear then we will need to linearise it or use a PECE method. Remember (from Numerical Methods II or equivalent) that stable solutions of such matrix systems requires a strictly diagonally dominant matrix. Strict diagonal dominance means

$$|a_{ii}| > \sum_{j=1, j\neq i}^{N} |a_{ij}|,$$

where a_{ij} is the entry in the ith row and jth column of A. Note: if $>$ is replaced by $\geq$ the system is diagonally dominant (not strict). Usually we get a good solution with diagonal dominance, but it is not guaranteed. For our problem strict diagonal dominance requires that

$$1 > h, \tag{3.24}$$

$$|1+\beta_{n-1}| > \sum_{i=0}^{n-2} |\beta_i|. \tag{3.25}$$

The step size h will need to be small enough for this.

The matrix system may be solved using a direct method or an iterative method; for large systems, the method should make use of the fact that the matrix is sparse. Methods for solving systems of linear equations should have been seen in Numerical Methods II. We will just use the Matlab command $\mathbf{y}^m = M \backslash \mathbf{y}^{m-1}$.

Example Determine the matrix system of equations for the previous example when solved using the implicit Euler method.

3.5.2 Direct finite-difference method for second-order IVPs

Consider

$$\frac{d^2x}{dt^2} = f(t, x, x'), \quad x(0) = c_0,\ x'(0) = c_1.$$

Using the central difference formula (error $\mathrm{O}(h^2)$) for x'' gives

$$\frac{x_{n+1} - 2x_n + x_{n-1}}{h^2} = f(t_n, x_n, x'_n)$$

$$\Rightarrow \quad x_{n+1} = 2x_n - x_{n-1} + h^2 f_n,$$

and where x' appears in the equation we substitute

$$x' \approx \frac{x_{n+1} - x_{n-1}}{2h},$$

which also has error $\mathrm{O}(h^2)$. The order of accuracy for the difference approximation of the first derivative should be as good as that for the second derivative.

We are given $x_0 = x(0) = c_0$. Also

$$x'(0) = c_1 \approx \frac{x_1 - x_{-1}}{2h} \quad \Rightarrow \quad x_{-1} = x_1 - 2hc_1.$$

Example

$$x''(t) + a(t)x'(t) + b(t)x(t) = r(t), \quad x(0) = c_0,\ x'(0) = c_1.$$

Use the centred finite difference approximations to determine a numerical solution method.

3.6 Numerical Solution of Boundary Value Problems

Until now we have only consider initial value problems. For completeness we consider second order boundary value problems. They can be solved numerically using direct finite-difference methods quite similar to those used for initial value problems. They can also be converted to initial value problems and solved using "shooting methods".

A boundary value problem differs from an initial value problem in that, instead of having two initial conditions, we specify the function x at two different 'times' t which are boundaries of the interval of interest. We usually think of the independent variable as a position rather than a time. Then we are solving an ODE for a function over a given spatial domain. Because of this we will use position x as the independent variable and $y(x)$ as the function for which we need to solve.

3.6.1 Direct finite-difference method for second-order BVPs

Dirichlet boundary conditions

A 2nd-order boundary value problem has the form

$$\frac{d^2y}{dx^2} = f(x, y, y'), \quad y(a) = \alpha,\ y(b) = \beta.$$

Here a and b are the boundaries of the computational domain $a \le x \le b$. In practice, however, we may be able to obtain values of y outside of this domain.

We proceed by approximating y'' and y' using finite-difference formulas, to transform the problem into a difference equation. As with solving an IVP, we should use difference formulas for y'' and y' that are of the same order of accuracy. From this we obtain a sparse system of equations which must be solved simultaneously for all the unknowns $y_i = y(x_i)$, where the x_i are grid points.

Example

Construct a numerical solution of the BVP

$$y''(x) + a(x)y'(x) + b(x)y(x) = r(x), \quad y(a) = \alpha,\ y(b) = \beta.$$

Neumann BCs

Suppose we use a Neumann (derivative) BC at $x = b$, i.e.

$$y'(b) = \beta.$$

How does this change the system of equations we solve? We no longer know $y_N = y(b)$ and so have another unknown in the problem — a total of N unknowns. We need N equations to solve for these. As we saw for the 2nd-order IVP, we may write

$$\frac{y_{N+1} - y_{N-1}}{h} = \beta \quad \Rightarrow \quad y_{N+1} = y_{N-1} + h\beta.$$

The (tridiagonal) system of equations to be solved is

$$\begin{aligned} -B_1 y_1 + A_1 y_2 &= D_1 - C_1\alpha, \\ C_n y_{n-1} - B_n y_n + A_n y_{n+1} &= D_n, \quad n = 2, \ldots, N-1, \end{aligned}$$

and

$$\begin{aligned} & C_N y_{N-1} - B_N y_N + A_N y_{N+1} = D_N \\ \Rightarrow \quad & (C_N + A_N) y_{N-1} - B_N y_N = D_N - A_N h\beta. \end{aligned}$$

3.6.2 Shooting methods for second-order BVPs

Now we write the BVP as a system of first-order ODEs. Putting $z_1 = y$, $z_2 = y'$:

$$z_1' = z_2, \quad z_1(a) = \alpha \tag{3.26}$$

$$z_2' = f(x, z_1, z_2), \quad z_1(b) = \beta. \tag{3.27}$$

We replace the BC at $x = b$ with a IC guess $z_2(a) = \gamma$ and solve the resulting IVP. For a nonlinear problem we will, in general, need to iterate, changing the IC $z_2(a) = \gamma$ at each iteration until we find the correct solution that satisfies the BC $z_1(b) = \beta$. There is no guarantee of convergence.

However for a linear problem we can use the superposition principle to determine the correct solution from just two guesses. The method is

1. Solve the IVP with $z_2(a) = \gamma_1$ to give solution $(z_1^{(1)}(x), z_2^{(1)}(x))$.
2. Solve the IVP with $z_2(a) = \gamma_2$ to give solution $(z_1^{(2)}(x), z_2^{(2)}(x))$.
3. The true solution is a linear combination of these:

$$z_1(x) = c_1 z_1^{(1)}(x) + c_2 z_1^{(2)}(x), \tag{3.28}$$
$$z_2(x) = c_1 z_2^{(1)}(x) + c_2 z_2^{(2)}(x), \tag{3.29}$$

   such that $z_1(a) = \alpha$ and $z_1(b) = \beta$. Hence

$$c_1 + c_2 = 1 \tag{3.30}$$
$$c_1 z_1^{(1)}(b) + c_2 z_1^{(2)}(b) = \beta. \tag{3.31}$$

This method can be extended to an nth order BVP and to Neumann and mixed BCs.

4 Two-dimensional autonomous ODE models

In this chapter we consider fixed points, stabilities and bifurcations in two-dimensional (second-order) ODE models, expressed as systems of two first-order ODEs with two unknown functions.

4.1 Existence and Uniqueness of Solutions

The Picard-Lindelöf theorem seen earlier can be extended to systems of ODEs.

Theorem 4.1. *Picard-Lindelöf theorem. Consider the nth order initial value problem*

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(t, \mathbf{x}), \tag{4.1}$$

$$\mathbf{x}(t_0) = \mathbf{x}_0, \tag{4.2}$$

where $\mathbf{f} = (f_1, f_2, \ldots, f_n)^T$. *Suppose* $I = [t_-, t_+]$ *is an interval in* t *with* $t_- < t_0 < t_+$ *and* $D \subset \mathbb{R}^n$ *with* $\mathbf{x}_0 \in D$. *If* $f_i : I \times D \to \mathbb{R}$ *is continuous on its domain and Lipschitz continuous on* D *for each* $i = 1, 2, \ldots, n$, *that is*

$$|f_i(t, \mathbf{x}) - f_i(t, \mathbf{y})| \leq L\|\mathbf{x} - \mathbf{y}\|,$$

then there exists $\epsilon > 0$ *for which the solution exists and is unique for* $t \in [t_0 - \epsilon, t_0 + \epsilon]$.

4.2 Qualitative Analysis of 2×2 Systems

Consider a system of two 1st-order ODEs

$$\frac{dx}{dt} = f(x, y), \quad \frac{dy}{dt} = g(x, y).$$

At each $\mathbf{x} = (x, y)$ of the *phase plane* we can plot

$$\mathbf{f}(\mathbf{x}) = (f(\mathbf{x}), g(\mathbf{x}))$$

to yield the *vector field*. A solution $\mathbf{x} = (x(t), y(t))$ represents a parametric curve in the (x, y) plane called a *trajectory* or an *orbit*, whose tangent vector $\mathbf{x}'(t)$ is specified by the vector field

$$\mathbf{f}(\mathbf{x}(t)) = (f(x(t), y(t)), g(x(t), y(t))).$$

It is sometimes convenient to consider only the direction of the vectors and not the magnitude. This yields a *direction field* for the system.

Since solution curves are tangential to the vector field, we often can follow trajectories just by following the arrows. Different solutions are obtained, corresponding to different initial conditions, by starting at a different point in the field.

The sketch of the (x, y) plane with a number of typical solutions is called a *phase portrait*.

Insight into the phase portrait may be obtained by considering the *nullclines*. The x-nullcline is the set of (x, y) points such that

$$x' = f(x, y) = 0,$$

i.e.

$$n_x := \{(x, y) \,|\, f(x, y) = 0\}.$$

Similarly the y-nullcline is

$$n_y := \{(x, y) \,|\, g(x, y) = 0\}.$$

On n_x all vectors are vertical and on n_y all vectors are horizontal. At intersections of the two nullclines we have a *steady state* or *equilibrium point*, since $x' = 0$ and $y' = 0$.

The steady states play an important role in the understanding of the whole dynamics. In many cases, if the behaviour near each steady state is known, then the global behaviour of solutions can be understood quite well. In fact, we can classify all possible behaviours which can occur near a steady state. We will do this for linear and non-linear systems.

4.3 Linear Systems

We begin by considering systems of the form

$$x' = \alpha x + \beta y \tag{4.3}$$
$$y' = \gamma y + \delta x. \tag{4.4}$$

4.3.1 Revision of linear algebra

Consider the n-dimensional linear system of equations, written in matrix form

$$\mathbf{x}' = A\mathbf{x}.$$

We look for solutions of the form $\mathbf{x} = e^{\lambda t}\mathbf{v}$ where $\mathbf{v}$ is a constant vector. Substituting into our matrix system of ODEs gives

$$\lambda e^{\lambda t}\mathbf{v} = Ae^{\lambda t}\mathbf{v} \quad \Rightarrow \quad \lambda\mathbf{v} = A\mathbf{v} \quad \Rightarrow \quad (A - \lambda I)\mathbf{v} = 0.$$

Thus λ is an eigenvalue of A and $\mathbf{v}$ the corresponding eigenvector. Hence, for an n-dimensional system of linear equations with n eigenvalues $\lambda_1, \ldots, \lambda_n$ and corresponding eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$, the solution is

$$\mathbf{x} = C_1 e^{\lambda_1 t}\mathbf{v}_1 + C_2 e^{\lambda_2 t}\mathbf{v}_2 + \ldots + C_n e^{\lambda_n t}\mathbf{v}_n.$$

4.3.2 Real eigenvalues

We start with the simplest linear system

$$x' = \alpha x,$$
$$y' = \gamma y,$$

which has the unique steady state $(x, y) = (0, 0)$ and the solution

$$x(t) = x(0)e^{\alpha t}, \quad y(t) = y(0)e^{\gamma t}.$$

The x-nullcline is $x = 0$ and the y-nullcline is $y = 0$.

Plotting the parametric curves $(x(t), y(t))$ for different initial values $(x(0), y(0))$ we find that there are 3 different phase portraits, depending on the signs of α, γ.

1. $\alpha > 0$, $\gamma > 0$ — all solutions diverge from the steady state $(0, 0)$ which is called a *source* or an *unstable node*.
2. $\alpha > 0$, $\gamma < 0$ — all solutions approach the x-axis and the steady state is called a *saddle*. (There is a qualitatively similar situation for $\alpha < 0$, $\gamma > 0$ when all solutions approach the y axis.)
3. $\alpha < 0$, $\gamma < 0$ — all solutions converge to the steady state $(0, 0)$ which is called a *sink* or a *stable node*.

4.3.3 Complex eigenvalues

Consider the linear system

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}.$$

For $\beta \neq 0$ the only steady state is the origin $(0, 0)$.

The x-nullcline is $\alpha x + \beta y = 0 \;\Rightarrow\; y = -\alpha x/\beta$.
The y-nullcline is $-\beta x + \alpha y = 0 \;\Rightarrow\; y = \beta x/\alpha$.

We can classify three qualitatively distinct cases:

1. $\alpha = 0$ and both eigenvalues are pure imaginary — all solutions are periodic and all trajectories are closed orbits surrounding the steady state $(0, 0)$, which is called a *centre*. These trajectories are called *periodic orbits*. The gravity pendulum is an example.
2. $\alpha > 0$ and both eigenvalues have positive real parts — all trajectories spiral away from the steady state $(0, 0)$ as time t increases; the steady state is an *unstable spiral* or a *spiral source*.
3. $\alpha < 0$ and both eigenvalues have negative real parts —all trajectories spiral towards the steady state $(0, 0)$ which is called a *stable spiral* or a *spiral sink*.

4.3.4 General Linear Systems

Next we come to the general linear system

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}, \quad A = \begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

$\det A \neq 0$. We make the transformation of coordinates

$$\begin{pmatrix} x \\ y \end{pmatrix} = P\begin{pmatrix} w \\ z \end{pmatrix},$$

where P is a 2×2 invertible matrix. Then

$$\frac{d}{dt}\begin{pmatrix} w \\ z \end{pmatrix} = P^{-1}AP\begin{pmatrix} w \\ z \end{pmatrix} = B\begin{pmatrix} w \\ z \end{pmatrix}.$$

It can be shown that B has the same eigenvectors as A (exercise), so that the systems

$$\mathbf{w}' = B\mathbf{w} \quad \text{and} \quad \mathbf{x}' = A\mathbf{x}$$

have the same phase portraits.

If A has two distinct real eigenvalues λ_1, λ_2 ($\lambda_1 \neq \lambda_2$), then we can choose P to diagonalise matrix A (the columns of P are the eigenvectors of A, i.e. $P = [\mathbf{v}_1, \mathbf{v}_2]$) giving

$$B = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

If A has two complex conjugate eigenvalues $\lambda_1 = \bar{\lambda}_2 = \alpha + \beta i$ (the corresponding eigenvectors are also complex conjugates), then we can choose P such that

$$B = \begin{pmatrix} \alpha & \beta \\ -\beta & \alpha \end{pmatrix}.$$

Here we write $\mathbf{v}_1 = \mathbf{a} + i\mathbf{b}$, with $\mathbf{a}$ and $\mathbf{b}$ real vectors, (then $\mathbf{v}_2 = \mathbf{a} - i\mathbf{b}$) and $P = [\mathbf{a}, \mathbf{b}]$.

Thus, the phase portraits of the general linear system will be the same as one of the two special cases considered earlier.

4.3.5 Asymptotic stability

Solutions only converge to the steady state $(0, 0)$ when both eigenvalues $\lambda_1, \lambda_2 < 0$ (the origin is a stable node), or when the real part of the eigenvalues satisfies $\alpha < 0$ (the origin is a stable spiral). When solutions converge to the steady state, we say that the steady state is *asymptotically stable*.

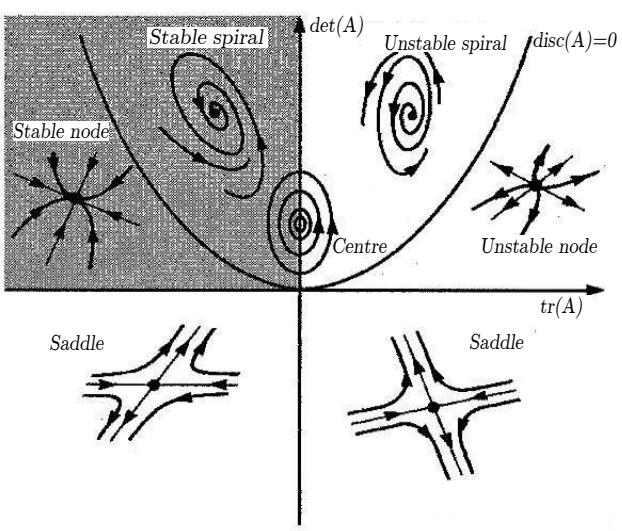

Fig. 4.1: The zoo for the general two-dimensional linear system (de Vries et al. 2006).

We have seen that we can classify the equilibria of a linear system according to the eigenvalues of the coefficient matrix

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

We can find the eigenvalues, λ_1, λ_2, of the matrix A using the trace, $\operatorname{tr} A = a + b$, and the determinant, $\det A = ad - bc$.

Theorem 4.2. *For a two-dimensional linear system* $\mathbf{x}' = A\mathbf{x}$*, the following are equivalent:*

- *the equilibrium* $(0, 0)$ *is asymptotically stable;*
- *all eigenvalues of* A *have negative real part;*
- $\det A > 0$ *and* $\operatorname{tr} A < 0$.

Fig. 4.1 shows the "zoo" of possible types of behaviour for steady states of two-dimensional systems.

4.4 Nonlinear Systems and Linearization

4.4.1 General nonlinear systems

We now come to a two-dimensional nonlinear system of equations

$$\begin{aligned} x' &= f(x, y), \\ y' &= g(x, y), \end{aligned}$$

where f and g are continuously differentiable functions.

Each pair $(\bar{x}, \bar{y})$ satisfying $f(\bar{x}, \bar{y}) = g(\bar{x}, \bar{y}) = 0$ is an equilibrium or steady state.

Definitions:

(a) A steady state $(\bar{x}, \bar{y})$ is called stable if a solution which starts nearby stays nearby.

(b) A steady state $(\bar{x}, \bar{y})$ which is not stable is called unstable (there is at least one solution which diverges from $(\bar{x}, \bar{y})$).

(c) A steady state $(\bar{x}, \bar{y})$ is asymptotically stable if $(\bar{x}, \bar{y})$ is stable, and all solutions near $(\bar{x}, \bar{y})$ converge to $(\bar{x}, \bar{y})$.

We can determine the stability of a steady state $(\bar{x}, \bar{y})$ by linearising the system about the steady state, as we did for discrete-time systems. Hence we write

$$\begin{aligned} x(t) &= \bar{x} + w(t), \\ y(t) &= \bar{y} + z(t), \end{aligned}$$

where $w(t)$ and $z(t)$ are assumed to be small perturbations to the steady state. Taking Taylor expansions

$$\begin{aligned} f(\bar{x} + w, \bar{y} + z) &= f(\bar{x}, \bar{y}) + w\frac{\partial f}{\partial x}(\bar{x}, \bar{y}) + z\frac{\partial f}{\partial y}(\bar{x}, \bar{y}) + \ldots, \\ g(\bar{x} + w, \bar{y} + z) &= g(\bar{x}, \bar{y}) + w\frac{\partial g}{\partial x}(\bar{x}, \bar{y}) + z\frac{\partial g}{\partial y}(\bar{x}, \bar{y}) + \ldots, \end{aligned}$$

or

$$\mathbf{f}(\bar{\mathbf{x}} + \mathbf{w}) = \mathbf{f}(\bar{\mathbf{x}}) + J(\bar{\mathbf{x}})\mathbf{w} + \text{higher order terms},$$

where J is the Jacobian matrix of $\mathbf{f}$ at the steady state $(\bar{x}, \bar{y})$. Note that $\mathbf{f}(\bar{\mathbf{x}}) = 0$ since $\bar{\mathbf{x}}$ is a steady state.

Also,

$$\begin{aligned} x' &= w', \\ y' &= z', \end{aligned}$$

so that, on dropping higher order terms, our system of equations becomes

$$\mathbf{w}' = J(\bar{\mathbf{x}})\mathbf{w} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \mathbf{w}.$$

This is a linear system and we know how to treat these. For most (but not all) steady states, conclusions obtained for the linearised system carry over to the original nonlinear system. The conditions under which this is true are given by the following definition and theorem.

Definition
The steady state $(\bar{x}, \bar{y})$ is called hyperbolic if all eigenvalues of the Jacobian $J(\bar{x}, \bar{y})$ have nonzero real part.

Theorem 4.3. ***The Hartman–Grobman Theorem.*** *Assume that $(\bar{x}, \bar{y})$ is a hyperbolic equilibrium. Then, in a small neighbourhood of $(\bar{x}, \bar{y})$, the phase portrait of the nonlinear system*

$$\begin{aligned} x' &= f(x, y), \\ y' &= g(x, y), \end{aligned}$$

is equivalent to that of the linearized system.

Remarks.

By Theorems 4.2 and 4.3, at a hyperbolic equilbrium $\bar{\mathbf{x}}$, stability properties are determined by the eigenvalues of the Jacobian matrix. This method of linearization may fail for nonhyperbolic equilibria.

The phrase "equivalent to" means that in a neighbourhood of $\bar{\mathbf{x}}$ there is a continuous one-to-one map between open sets that maps the vector field of the nonlinear system to the vector field of its linearization. In that case the phase portrait near the stationary point is one of those shown in Fig. 4.1.

4.5 Application to models

4.5.1 Interaction model for two populations

We consider two interacting populations $x(t)$ and $y(t)$. For each population on its own, we suppose that the population growth is exponential, and that interactions between two populations are proportional to $x\,y$. Therefore, the model is

$$\begin{aligned} \frac{dx}{dt} &= \alpha x + \beta xy, \\ \frac{dy}{dt} &= \gamma y + \delta xy, \end{aligned}$$

where α, β, γ, δ are constant real parameters.

There are 4 constants and each can have two signs. This gives 10 qualitatively different cases, which are summarised in Table 4.1.

In order to analyse the model using the general theory from §4.4, we express it as

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f(x, y) \\ g(x, y) \end{pmatrix},$$

with $f(x, y) = \alpha x + \beta xy$, $g(x, y) = \gamma y + \delta xy$.

The x-nullcline, n_x, is given by $f = 0$, i.e.

$$x = 0 \quad \text{or} \quad y = -\frac{\alpha}{\beta}.$$

Table 4.1: Classification of general 2-species interaction model.

α	β	γ	δ	
+	+	+	-	Predator (x) – prey (y) models
+	+	-	-	
-	+	+	-	
-	+	-	-	
+	+	+	+	Mutualism or symbiosis models
+	+	-	+	
-	+	-	+	
+	-	+	-	Competition models
+	-	-	-	
-	-	-	-	

The y-nullcline, n_y, is given by $g = 0$, i.e.

$$y = 0 \quad \text{or} \quad x = -\frac{\gamma}{\delta}.$$

The steady states are intersection points of the nullclines, satisfying $f = 0$ and $g = 0$. There are two:

$$P_1 = (0, 0), \quad P_2 = \left(-\frac{\gamma}{\delta}, -\frac{\alpha}{\beta}\right).$$

The linearised problem is

$$\frac{d}{dt}\begin{pmatrix} w \\ z \end{pmatrix} = J(\bar{x}, \bar{y})\begin{pmatrix} w \\ z \end{pmatrix} = \begin{pmatrix} \alpha + \beta\bar{y} & \beta\bar{x} \\ \delta\bar{y} & \gamma + \delta\bar{x} \end{pmatrix}\begin{pmatrix} w \\ z \end{pmatrix}.$$

First consider $P_1 = (0, 0)$.

$$J(0, 0) = \begin{pmatrix} \alpha & 0 \\ 0 & \gamma \end{pmatrix}$$

which has two eigenvalues $\lambda_1 = \alpha$, $\lambda_2 = \gamma$.

Next consider $P_2 = \left(-\frac{\gamma}{\delta}, -\frac{\alpha}{\beta}\right)$.

$$J\left(-\frac{\gamma}{\delta}, -\frac{\alpha}{\beta}\right) = \begin{pmatrix} 0 & -\beta\gamma/\delta \\ -\alpha\delta/\beta & 0 \end{pmatrix} = A.$$

We have $\operatorname{tr} A = 0$ and $\det A = -\alpha\gamma$, i.e. eigenvalues $\lambda_1 = \sqrt{\alpha\gamma}$, $\lambda_2 = -\sqrt{\alpha\gamma}$.

To go further we need more information, in particular, the signs of the parameters.

4.5.2 The Kermack-McKendrick epidemic model

A well-known model of the spread of an infectious disease is the SIR model (Fig. 4.2). There are both discrete and continuous forms

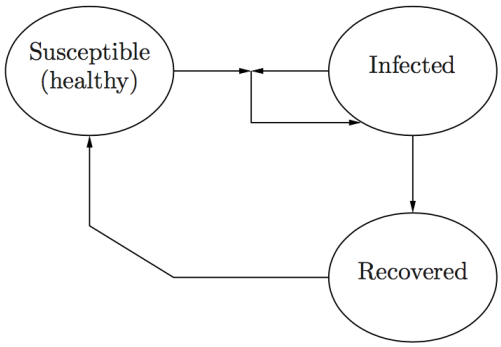

Fig. 4.2: Arrow diagram for a simple epidemic model, showing the relationships between the classes of susceptible, infected, and recovered individuals. Diagram taken from de Vries et al. (2006).

of this model. Here we consider the continuous form which uses ODEs rather than discrete-time equations. S, I and R are the sizes of the susceptible, infected and recovered groups, respectively. The general model is

$$\begin{aligned} \frac{dS}{dt} &= -\beta IS + \gamma R, \\ \frac{dI}{dt} &= \beta IS - \alpha I, \\ \frac{dR}{dt} &= \alpha I - \gamma R. \end{aligned}$$

We will assume $\gamma = 0$, so that the model simplifies to

$$\begin{aligned} \frac{dS}{dt} &= -\beta IS, \\ \frac{dI}{dt} &= \beta IS - \alpha I. \end{aligned}$$

This is the Kermack-McKendrick model.

Exercise: show that this model can be scaled to yield a dimensionless model with no parameters.

Steady states of the Kermack-McKendrick model are $I = 0$, $S \geq 0$, i.e. we have a *ray* of steady states extending along the positive S axis. Fig. 4.3 shows the phase portrait for this system.

Biologically, the phase portrait reveals an important fact in epidemiology: α/β represents the critical population size to sustain an epidemic. If the initial susceptible population is below α/β then no epidemic is possible — the number of infections decreases in time. If $S(0) = S_0 > \alpha/\beta$ then we have an epidemic, with the number of infections first increasing, then reaching a maximum when $S = \alpha/\beta$, and then declining.

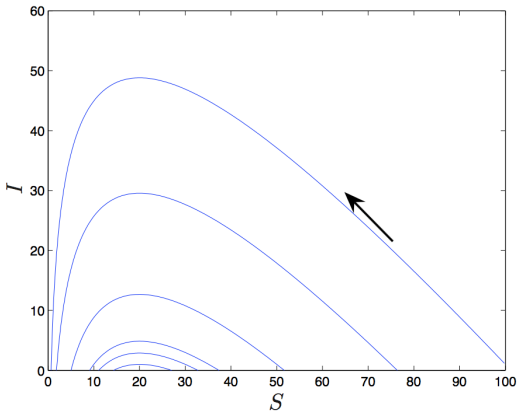

Fig. 4.3: The phase portrait for the epidemic model. Case with $\alpha/\beta = 20$. The arrow shows the direction of travel.

4.6 Limit cycles

Consider the system

$$\dot{r} = r(1 - r^2), \quad \dot{\theta} = 1, \tag{4.5}$$

where $r \geq 0$ and θ are *polar coordinates*. The radial and angular dynamics are uncoupled and so can be analysed separately.

The steady states of the first ODE are $r^* = 0, 1$, where $r^* = 0$ is an unstable steady state and $r^* = 1$ is stable steady state. However, $\theta = t + \theta_0$, is changing continuously in time. Back in the phase plane, all trajectories (except $r^* = 0$) spiral asymptotically toward the unit circle $r^* = 1$, which is a *limit cycle*.

A limit cycle is an *isolated* closed trajectory, or periodic orbit. Neighbouring trajectories spiral either towards or away from the limit cycle, depending on whether it is stable or unstable. A stable limit cycle is also said to be *attracting*.

We can solve this problem exactly. Already we have $\theta = t + \theta_0$. The ODE for r is separable and we find

$$r = \frac{r_0 e^t}{\sqrt{1 - r_0^2 + r_0^2 e^{2t}}}.$$

Some trajectories are show in Fig. 4.4

Limit cycles arise in models of real-world problems and there is a considerable theory associated with them that we do not have time to consider here.

4.7 Bifurcations in 2D systems

In going from 1D to 2D we still find that fixed points can be created or destroyed or destabilised as parameters are varied — but now the

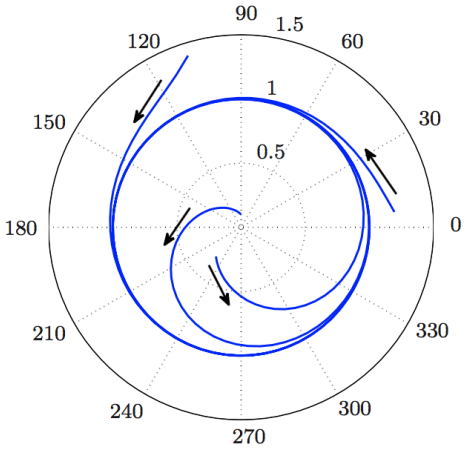

Fig. 4.4: Some trajectories in the (r, θ) phase plane for the system (4.5).

same is true of closed orbits as well. In 2D we say that a bifurcation has occurred if the phase portrait changes its topological structure as a parameter is varied.

4.7.1 Saddle-node, transcritical and pitchfork bifurcations

The bifurcations of fixed points seen in 1D have analogs in 2D. Nothing new really happens when when the second dimension is added — all the action is confined to a 1D subspace along which the bifurcations occur, while in the extra dimension there is simple attraction or repulsion from that subspace.

The canonical forms of these bifurcations in 2D are

$$\begin{aligned}
\text{Saddle-node bifurcation:} \quad & \dot{x} = \mu - x^2, \quad && \dot{y} = -y, \\
\text{Transcritical bifurcation:} \quad & \dot{x} = \mu x - x^2, \quad && \dot{y} = -y, \\
\text{Supercritical pitchfork bifurcation:} \quad & \dot{x} = \mu x - x^3, \quad && \dot{y} = -y, \\
\text{Subcritical pitchfork bifurcation:} \quad & \dot{x} = \mu x + x^3, \quad && \dot{y} = -y.
\end{aligned}$$

4.7.2 Hopf bifurcation

A new type of bifurcation, known as a Hopf bifurcation, arises in 2D systems. Consider the system

$$\begin{aligned}
\dot{x} &= -y + x(\mu - x^2 - y^2), \\
\dot{y} &= x + y(\mu - x^2 - y^2),
\end{aligned}$$

with $(x, y) \in \mathbb{R}^2$, $\mu \in \mathbb{R}$.

Using polar coordinates $x = r\cos\theta$, $y = r\sin\theta$, $r \geq 0$, we can write this system (exercise) as

$$\begin{aligned}
\dot{r} &= r(\mu - r^2) \\
\dot{\theta} &= 1.
\end{aligned}$$

We've seen this system before, with $\mu = 1$, when looking at limit cycles. Note that the equation for r is the normal form for a pitchfork bifurcation, i.e. as μ passes through the bifurcation value 0 it undergoes a pitchfork bifurcation.

The steady state $r^* = 0$ corresponds to the steady state $(x^*, y^*) = (0, 0)$ of the original system.

The steady state $r^* = \sqrt{\mu}$ corresponds to $x^2 + y^2 = \mu$, a periodic orbit. (Note, $r^* = -\sqrt{\mu}$ doesn't make sense here.)

To visualise what is going on we plot phase portraits in the (x, y) or (r, θ) plane; refer again to Fig. 4.4.

For $\mu < 0$, r decreases with t while θ increases — a spiral sink (not shown in Fig. 4.4).

For $\mu > 0$ and $r_0^2 > \mu$, r decreases with t while θ increases until the periodic orbit $r^2 = \mu$ is reached.

For $\mu > 0$ and $r_0^2 < \mu$, r increases with t while θ increases until the periodic orbit $r^2 = \mu$ is reached.

The corresponding bifurcation diagram may be plotted in (μ, x, y) or (μ, r, θ) space. As μ increases through 0, the branch of steady states at the origin, given by $(x^*, y^*) = (0, 0)$ loses its stability and a branch of stable periodic orbits emerges. This is a *Hopf bifurcation.*

Note that for this system we can compute

$$J(0, 0) = \begin{pmatrix} \mu & -1 \\ 1 & \mu \end{pmatrix},$$

which has a pair of complex eigenvalues $\lambda = \mu \pm i$. At the bifurcation value $\bar{\mu} = 0$ these are purely imaginary. The occurrence of purely imaginary eigenvalues for a set of parameter values is an important indicator for Hopf bifurcation.

Bibliography

M Abramowitz, IA Stegun (1965) *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*. National Bureau of Standards, Washington DC.

JC Butcher, *Numerical Methods for Ordinary Differential Equations*, Wiley, West Sussex, England, 2003.

G Dahlquist and A Björck, *Numerical Methods*, Dover Publications, Inc., Mineola, New York, 2003.

G de Vries, T Hillen, M Lewis, J Müller, B Schönfisch, *A Course in Mathematical Biology*, SIAM 2006.

E Kreyszig (1978) *Introductory Functional Analysis with Applications*. John Wiley & Sons.

RMM Mattheij, SW Rienstra and JHM ten Thije Boonkkamp, *Partial Differential Equations: modelling, analysis, computation*, SIAM, Philadelphia, 2005.

SS Rao, *Applied Numerical Methods for Engineers and Scientists*, Prentice Hall, 2002.

SH Strogatz (2000) *Nonlinear Dynamics and Chaos with Applications to Physics, Biology, Chemistry, and Engineering*, Perseus Publishing, Cambridge, MA.