

MA30287: Mathematics of Planet Earth

Philippe H. Trinh

Last updated: 26 March 2025

Mathematical Sciences
University of Bath

Table of contents

Table of contents	ii
List of tables	iv
List of figures	iv
Lecture plan	x
I Introduction	1
1 What does Mathematics of Planet Earth mean?	2
2 Conservation laws	4
3 Dimensional scaling analysis	10
4 Basic energy models	14
II Practical applied mathematics	19
5 Asymptotic approximations I	21
6 Asymptotic approximations II	26
7 Numerical solutions of IVPs	30
8 Asymptotic approximations III	33
9 Nonlinear root finding	38
10 Numerical solutions of PDEs	42
11 An example of a wine cellar problem	44
12 The wine cellar problem	45
III Energy balance models	48
13 EBM with nonlinear albedo	49
14 EBM with latitude I	54

15 EBM with latitude II	58
16 EBM with latitude III	64
17 EBM with latitude IV	68
IV Fast-slow dynamics	72
18 Ice ages I: an introduction	74
19 Ice ages II: a model for fast-slow dynamics	79
20 Ice ages III: analysis of the van der Pol equation	82
21 Fast-slow dynamics for higher-order systems	87
V Box models of the ocean and environment	92
22 Basic models of the ocean	93
23 Stommel's box model	100
24 Box models for flood estimation	107
VI Exercises	113
25 Problem set 1	114
26 Problem set 2	117
27 Problem set 3	120
28 Problem set 4	123
29 Problem set 5	127
30 Problem set 6	130
31 Problem set 7	132
32 Problem set 1 solutions	134
33 Problem set 2 solutions	140
34 Problem set 3 solutions	146
35 Problem set 4 solutions	150
36 Problem set 5 solutions	158
37 Problem set 6 solutions	163
38 Problem set 7 solutions	169

VII	Problem classes	173
39	Problem class 1: an introduction to Noteable	174
40	Problem class 2: dimensional analysis	178
41	Problem class 3: BVPs	183
42	Problem class 4: PDEs and numerical stability	184
43	Problem class 5	186
44	Problem class 6	187
VII	Appendices	188
45	Differential equations	189
46	Dynamical systems	191
47	Vector calculus	193
48	Modelling	194
49	Finite difference approximations	195
50	Coding	199
51	An example of ChatGPT going wrong	206
	References	211

List of tables

List of figures

1	This picture was produced by Abstruse Goose and has the title: “this is how scientists see the world”.	vii
1.1	The many components of a full Earth System Model	2

4.1	Radiation in the atmosphere	15
4.2	An illustration of Fourier's glass box	17
12.1	Numerical solution of the wine cellar problem	46
13.1	Bifurcation diagram of Q/Q_0 vs T_∞	51
15.1	The four possible solutions; orange and blue show the completely-ice and ice-free states; red and green show the partial ice-covered state. These are shown for $A = 202$, $B = 1.9$, $k = 1.6B$, and $Q = 342\$$	63
16.1	The iceline	64
16.2	Mean temperature	65
16.3	From Cahalan & North, p.1179	66
16.4	Bifurcation diagram for (Q, \bar{T})	67
18.1	Figure from here	74
18.2	Figure from here	75
18.3	Figure from here	76
19.1	Phase plot of the above system	81
20.1	Phase plot of the above system	82
20.2	Illustration of dynamics along the slow manifold	83
20.3	Fast dynamics	85
21.1	Numerical calculation of the system of three equations with $\lambda = 1$, $\mu = 2$, and $\epsilon = 0.1$	88
21.2	Plot of the solution evolution with time	89
21.3	Numerical solution as before but now plotted in the (c, s) phase plane. The dashed line is the approximation derived above.	90
21.4	Numerical solution of the single differential equation	91
22.1	The thermohaline circulation. In the Atlantic, the circulation carries warm water (red) north near the surface and cold deep water (blue) south. Image from NASA/JPL.	94
22.2	Cross section of the Atlantic Ocean, showing the temperature and salinity profiles on the right. Image from (Kaper und Engler, 2013).	95
22.3	Image from (Kaper und Engler, 2013)	95
22.4	Two-box model of the North Atlantic with evaporation and precipitation. Image from (Kaper und Engler, 2013)	97
23.1	Phase plane for $R = 2$, $\lambda = 1/5$ and $\delta = 1/6$. The blue, red, and green fixed points correspond to f_1 , f_2 , and f_3 in order. The blue point is a stable node, the red point is an unstable saddle, and the green point is a stable spiral.	104
23.2	105
24.1	Table of flood damage; UK government report given in reference above . . .	107
24.2	Typical geometry of consideration; image from Gauch and Lin (2020) . . .	108
24.3	Calibration and testing for model 1	112
24.4	Zoom of the above	112
24.5	Calibration and testing for model 2	112
24.6	Zoom of the above	112
33.1	Hand drawn image	141

33.2 Python image	142
33.3 With (solid) and without damping (dashed)	144
34.1 A plot of the above data. This looks linear, and indeed this provides numerical confirmation that the boundary layer should be sized with ϵ	147
35.1 Graphical solution for Q3a	155
35.2 Graphical solution for Q3b	156
35.3 Graphical solution for Q3c	157
36.1 Graphical solution	158
36.2 Graphical solution	159
36.3 Graphical solution	159
36.4 Graphical solution	160
36.5 Graphical solution	160
37.1 Illustration of slow manifold	164
38.1 Steady state intersections for Stommel's box model	170
38.2 Bifurcation diagram for R vs f^*	171
46.1 Classification diagram from (Kaper und Engler, 2013)	192
50.1 What ChatGPT says on scope	199
50.2 A nice chat with ChatGPT	201
51.1 Example ChatGPT output	207
51.2 Example ChatGPT output	208
51.3 Not even wrong	209

Preface

Note 30 Jan 2025

We are currently re-generating the notes for 2025 delivery. If you would like to see the complete notes from prior deliveries there should be a link up on the Moodle site. Hang tight.

Welcome to the 2023-24 delivery of **MA30287 Maths of Planet Earth** at the University of Bath. In the below figure, you will see an image that, to some extent, represents the way mathematicians and other learned scientists see the Planet.

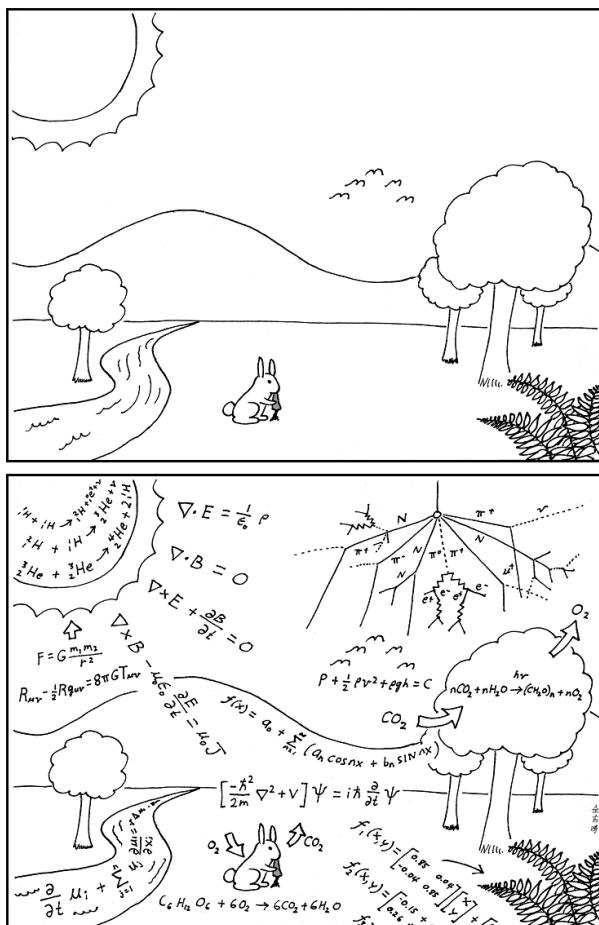


Figure 1: This picture was produced by **Abstruse Goose** and has the title: “this is how scientists see the world”.

Lectures and office hours

Lectures take place at the following times and locations:

- Tuesdays 9:15 in 6W 1.2
- Wednesdays 11:15 in 6W 1.2
- Thursdays 15:15 in 6W 1.2

Office hours: You will be able to find me for an office hour in 4W 2.18 on Thursdays (following the lecture). Typically it is best to set this up, beforehand, by email appointment.

Coursework and examinations

Your final mark will be 25% coursework and 75% final exam.

Details of the coursework will be released in Week 7 and it will be due in Week 10¹.

Resources

In general, you will have access to a few kinds of resources:

1. The [Moodle portal](#) will be the main organisation portal.
2. Lecture notes, coursework, and other resources will be found in an online format and will be linked on Moodle.
3. Coding will be done via Noteable, accessed via the Moodle website.

Naturally, because this is a relatively new module at Bath, there will be a fair amount of activity as we settle the material over the semester.

Prerequisites

This course officially requires MA20221 (modeling and dynamical systems) or XX20231 (mathematical and statistical methods for the life sciences). It is designed to be somewhat stand-alone in the sense that applied mathematical techniques learned in other courses will be introduced in some capacity. Such techniques will involve:

- Solutions of ordinary differential equations (MA10230 and MA20220).
- Multivariable calculus, partial differentiation, and multiple integrals (MA10230 and MA10236); some review/introduction of concepts from MA20223.
- Dynamical systems, stability, phase planes (MA20221, MA30060).
- Numerical methods in Python (MA10276).

Whenever possible, I have isolated such reviews/introductions and these can be found in the *Mathematical methods* section of these notes. During the current phase, Bath is undergoing changes to Years 1 and 2, so some of the above content may have shifted; these shifts are typically minor and cosmetic.

¹Subject to confirmation.

Resources

This course is designed around the following sources:

- Sustainable energy – without the hot air by ([MacKay, 2009](#))
- Mathematics & Climate by ([Kaper und Englert, 2013](#))
- Mathematical Geoscience by ([Fowler, 2011](#))
- [A gentle introduction to numerical simulations with Python](#)

However, the current MPE course is also quite ‘bespoke’ at Bath, so there is no single source with all the information. It has been designed to cover a selection of topical subjects and complementary to other materials you might have learned previously in your studies.

Lecture plan

A lecture plan can be found at [this link](#).

Part I

Introduction

CHAPTER 1

What does Mathematics of Planet Earth mean?

Mathematics of Planet Earth seems like an incredibly broad description for a course, but perhaps in order to give a rough idea of what such a course might include, we can consider the following diagram in Figure 1.1, which illustrates different categories and subject areas that are involved in the modelling of a full Earth system.

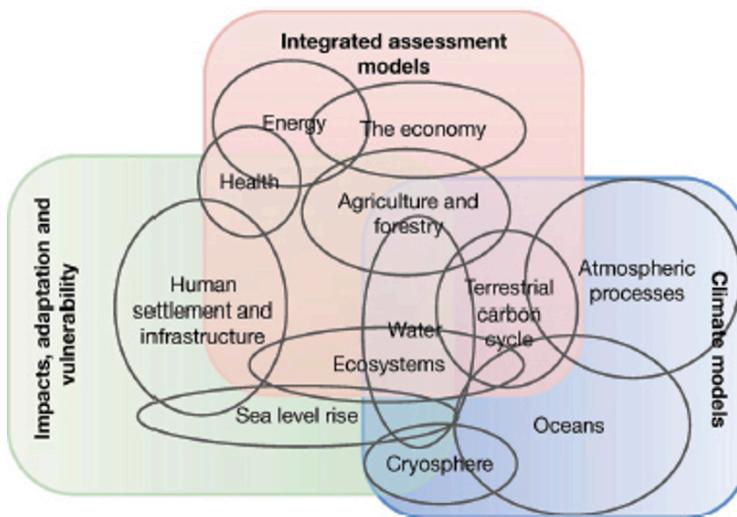


Figure 1.1: The many components of a full Earth System Model

It would be possible to spend a lifetime studying any one aspects of the above categories, and they span many different areas of study, including: (i) engineering (civil, fluids, mechanical, etc.); (ii) physics (geosciences, mechanics); (iii) Earth sciences; (iv) policy and health; and so forth and so on. As mathematicians, we also have a unique perspective, and applied mathematics plays important roles in many of the above categories.

In essence, this course will include topics and themes are united by aspects of **mathematical modelling** and **mathematical analysis** and this is what distinguishes our style of study from adjacent areas of science and social science.

This course will focus on **models** of Planet Earth. As such, one can divide modelling into (at least) three types: (i) conceptual modelling; (ii) physics-based modelling; and

(iii) statistical or data-based modelling. A basic **conceptual model** of the temperature on the planet consists of a differential equation that expresses basic principles of energy conservation, but that significantly coarse-grains the dynamics (to the extent where you would not necessarily need to know very much fluid mechanics or physics to study it). Analysis of such models is done numerically or semi-analytically (similar to the kind of phase-plane analysis of ODEs you would have seen in other modules). In this course, we will also study conceptual models of oceans and floods.

A secondary part of this course will involve more in-depth analysis of the physical models that govern the blue elements of the above figure. This moves us from the toy box (conceptual) models studied above to digging into the underlying physics—this also falls into the category of Mathematical Geoscience or Fluid Mechanics.

Statistical or data-based approaches are not as emphasised in this course (though they remain very important tools). Such approaches include, for example, analyses of time-series of climate variables, or machine-learning approaches for building data-based models on large quantities of data.

CHAPTER 2

Conservation laws

In the first chapter of (Fowler, 2011), there is a concise introduction to the different categories of techniques and approaches that you might use when doing mathematical modelling in the real world. Some of these ideas will be introduced to you in this course.

Here, we provide a brief intro to the highlights, involving the use of conservation laws (and PDEs) and also the concept of non-dimensionalisation (which you would have encountered previously), studied in Chapter 3.

Partial differential equations

Knowledge partial differential equations (PDEs) is not a prerequisite for this course, but naturally in studying anything related to the physical real world, we must discuss such things. The hope is that the necessary theory for PDEs will be presented to you as this course evolves, so that it can appreciated by both newcomers and experienced readers.

Conservation laws can be expressed as mathematical equations that represent the idea that some quantity is conserved. In processes governing the planet, these might correspond to conservation of heat, of water, of air, of momentum, etc.

In Chapter 4, we will develop the simplest possible model governing the temperature on the surface of the Earth. It is a conservation equation for energy and is zero-dimensional (does not involve time and does not involve spatial variation).

2.1 Derivation of the 1D heat equation

In order to demonstrate some of the basic principles of this course, let us demonstrate the derivation of the heat equation. We are interested in modelling the heat in a volume, V , which, for the sake of concreteness is given by a long cylinder with its axis along $x \in [0, L]$. We assume that the side walls of the cylinder are insulated and the temperature only varies along the x direction.

At any point along this rod, the internal heat is given by $\rho c T(x, t)$, where ρ is the density of the material (kg/m^3), c is the specific heat capacity ($\text{J}/(\text{kg K})$), and T is the temperature (K). Therefore, the heat energy along any segment in the rod is calculated from

Internal heat energy

$$\text{heat energy in } [a, b] = \int_a^b \rho c T \, dx.$$

If the heat changes, then the rate of change of heat energy is given by the time derivative of the above quantity. By conservation of energy, any change of the internal energy must be equal to the inflow or outflow of heat at the ends, $x = a$ or $x = b$. We therefore write q for the flux (or flow) of heat.

We need a **constitutive law** that dictates how energy is exchanged at the boundaries. Based on intuition, it is sensible to assume that the flow of heat proceeds from hot to cold. For example, hot air rises towards cool air; or heat from a hot mug of tea flows and diffuses outwards into a cold room. Therefore, we write this as

Fourier's law

Fourier's law in 1D specifies that the heat flux is given by

$$q(x, t) = -k \frac{\partial T}{\partial x}.$$

This is known as Fourier's law. The quantity k is the thermal conductivity, and its units are $\text{W}/(\text{m K})$. Because a Watt is a Joule/s, you can also see that the units of k are $\text{J}/(\text{m K s})$. The quantity q is the flux, and you can verify that it is given in units of $\text{J}/(\text{m}^2 \text{s})$.

Therefore by energy conservation, we have

$$\frac{d}{dt} \int_a^b \rho c T \, dx = q(x = a, t) - q(x = b, t),$$

i.e. the change in internal heat is equal to the flow through the ends. Note that we have added-in the heat flux due to $x = a$ assuming positive q refers to heat moving from left-to-right. Conversely, we subtract it away from $x = b$. We can alternatively write this as

$$\frac{d}{dt} \int_a^b \rho c T \, dx = - \int_a^b \frac{\partial q}{\partial x} \, dx.$$

Substitution Fourier's law, we can then write

$$\int_a^b \rho c \frac{\partial T}{\partial t} \, dx = \int_a^b k \frac{\partial^2 T}{\partial x^2} \, dx.$$

Because the above integral identity needs to be true for all possible values of a and b , then it must be true everywhere (this is sometimes referred to as the *du Bois-Reymond lemma* or the *bump lemma*). Therefore we are left with the classic heat equation.

Heat equation

$$\rho c \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2}. \quad (2.1)$$

In order to produce a sensible physical solution, partial differential equations are typically supplemented by initial conditions and boundary conditions. The **initial condition**

prescribes the state of the function at some initial time, typically $t = 0$. **Boundary conditions** prescribe how the function behaves on the boundary of its domain, which in this case is $x = 0$ and $x = L$. An example might be

Initial conditions (IC) and boundary conditions (BC)

$$T(x, 0) = T_0$$

$$T(0, 0) = T_a$$

$$T(L, 0) = T_b$$

which expresses, respectively, that the temperature starts from a constant temperature, T_0 , and where the ends of the rod are kept at temperature T_a and T_b .

Steady states and long-time behaviours

When we refer to a **steady-state** solution, we are typically referring to a time-independent solution.

Definition 2.1 (Steady-state solutions). Given an evolving system described by a function, say $f(x, t)$, defined on some spatial domain and with $t > 0$, the steady-state solution refers to time-independent solutions with

$$\frac{\partial f}{\partial t} = 0.$$

One can envisage that as the system evolves with $t \rightarrow \infty$, it reaches a state that is independent of time. However, not all systems will approach a steady state. Moreover, not all steady states are stable or attractive (and might never be reachable in a real-life experiment).

For the case of heat flow, such a steady-state solution would be $T(x, t) = T(x)$. In this case,

$$\frac{\partial T}{\partial t} = 0 \implies k \frac{\partial^2 T}{\partial x^2} = 0.$$

Therefore, for the heat flow along a segment of length L with left boundary held at T_a and right boundary held at T_b we have

$$T_{\text{steady}}(x) = \left(\frac{T_b - T_a}{L} \right) x + T_a.$$

2.2 Deriving the 1D transport (continuity) equation

Consider the mass transport of some substance with density $\rho(x, t)$, immersed in a fluid, along a one-dimensional line in x . If desired, you may consider the substance as existing in three-dimensional space, and propagating along the x -direction, with its behaviour independent of y and z . Initially, when considered at time t , the mass of the substance between two points, a , and b , is:

$$m_{\text{blob}}(t) = \int_{a(t)}^{b(t)} \rho(x, t) \, dx.$$

As time increases the particles of the substance will move due to the fluid moving; at the same time, the fluid volume which is initially contained in $x \in [a(0), b(0)]$ will also

move. We want to find how the mass of the blob changes in time, and hence consider the quantity

$$\frac{dm_{\text{blob}}}{dt} = \int_a^b \frac{\partial \rho}{\partial t} dx + \rho(b, t) \frac{db}{dt} - \rho(a, t) \frac{da}{dt}.$$

The above considers the intrinsic rate of change of the function within the integrand, but then adds the extra mass due to the right boundary shifting rightwards (b) and subtracts the mass due to the left boundary shifting rightwards (a). It is known as the [Leibniz integral rule](#). We can thus write this within the integral as

$$\frac{dm_{\text{blob}}}{dt} = \int_a^b \left[\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} \left(\rho(x, t) \frac{dx}{dt} \right) \right] dx.$$

However, the quantity

$$\frac{dx}{dt} \equiv u(x, t)$$

represents the velocity of the fluid (which for the moment we assume to be a known and provided quantity). Therefore we can write the mass change as

$$\frac{dm_{\text{blob}}}{dt} = \int_a^b \left[\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} (\rho u) \right] dx.$$

This result, which explains how to pass a derivative through an integral express mass of a substance within a flow is known as the *Reynolds Transport Theorem*. We have just derived it in 1D.

Reynolds Transport Theorem

Let $\rho = \rho(x, t)$ be some quantity (such as density) that is advected along a one-dimensional line in x due to a fluid with velocity $u(x, t)$. Then

$$\frac{d}{dt} \int_{a(t)}^{b(t)} \rho dx = \int_{a(t)}^{b(t)} \left[\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} (\rho u) \right] dx.$$

We may now consider the substance being transported along the x -direction. If there is no interior creation or destruction of the source, then by conservation of mass, it must be the case that

$$\frac{dm_{\text{blob}}}{dt} = 0.$$

Thus, again since the above integral identity applies to all possible values of a and b , it must be the case that the integrand is zero. Thus we conclude with the so-called transport equation.

One-dimensional transport equation

The transport of a substance described by $\rho(x, t)$ advected along a one-dimensional line in x due to a fluid with velocity $u(x, t)$ is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} (\rho u) = 0.$$

Again, we must consider the above problem in combination with potential initial conditions and boundary conditions. For instance, we might specify that the substance begins from some initial state, say

$$\rho(x, 0) = \rho_0(x).$$

The boundary conditions are more subtle. It is not always obvious what the boundary conditions should be on a problem.

We can develop a solution of the above problem via a special coordinate substitution that takes into account the fact that solutions of the problem are wave-like. Let us set

$$\xi = x - ct,$$

and perform a substitution of the independent variable: $\rho(x, t) = u(\xi, t)$. Then by the chain rule,

$$\begin{aligned}\frac{\partial \rho}{\partial x} &= \frac{\partial \xi}{\partial x} \frac{\partial u}{\partial \xi} = \frac{\partial u}{\partial \xi}, \\ \frac{\partial \rho}{\partial t} &= \frac{\partial u}{\partial t} + \frac{\partial \xi}{\partial t} \frac{\partial u}{\partial \xi} = \frac{\partial u}{\partial t} - c \frac{\partial u}{\partial \xi}.\end{aligned}$$

Substitution into the PDE now gives

$$\frac{\partial u}{\partial t} = 0 \implies u = f(\xi),$$

where f is any arbitrary function. Therefore we claim that the general solution of the transport equation (with constant speed) is

$$u(x, t) = f(x - ct).$$

Here is an example of a solution of such a problem.

Linear advection equation with constant speed

Solve the problem given by

$$\frac{\partial \rho}{\partial t} + c \frac{\partial \rho}{\partial x} = 0,$$

where the velocity $u = c$ is constant. You may assume the initial condition is given by

$$\rho(x, 0) = \rho_0(x) = e^{-x^2}.$$

Solution

The general solution is given by

$$\rho(x, t) = f(x - ct),$$

where F is an arbitrary (differentiable) function. You can verify this via the chain rule, noting that

$$\frac{\partial f}{\partial t} = -cf'(x - ct) \quad \text{and} \quad \frac{\partial f}{\partial x} = f'(x - ct).$$

Applying the initial condition at $t = 0$ gives $f(x) = \rho_0(x)$ and thus

$$\rho(x, t) = e^{-(x-ct)^2}.$$

You may plot this to discover that it is, as perhaps expected, a Gaussian profile moving to the right at constant speed (for $c > 0$).

CHAPTER 3

Dimensional scaling analysis

Putting a mathematical model into non-dimensional form is fundamental (Fowler, 2011). When we refer to a “back-of-the-envelope calculation”, we often mean simple algebraic calculations that still nevertheless provide enormous insight on problems. Such analyses are often based on dimensional analysis, which identifies the relationships between the different quantities involved in the problem.

3.1 Dimensional quantities

Every physical quantity, say Q , can be expressed as a product of a dimensional unit, denoted $[Q]$, and a magnitude, say Q' . Thus we write

$$Q = Q'[Q]$$

For example, if x corresponds to the physical length in a problem, we might select $[x] = \text{km}$ or $[x] = \text{yards}$ or $[x] = \text{m}$. It is important to choose the dimensionalisation to suit the problem under consideration.

SI Units

The International System (SI) of Base Units sets out a distinct selection of choices for dimensions in certain physical quantities. The seven fundamental dimensional units are

- [Length] = metre, m
- [Time] = seconds, s
- [Mass] = kilogram, kg
- Temperature = Kelvin, K,
- [Electric current] = ampere, A
- [Light intensity] = candela, cd
- [Material quantity] = mole, mol

Dimensional units that can be expressed in terms of other fundamental units are known as *derived units*. For example:

- [Speed] = m s^{-1}
- [Acceleration] = m s^{-2}
- [Force] = kg m s^{-2}

3.2 Dimensional homogeneity and non-dimensionalisation

All terms in any equation must have the same dimensions. This is the principle of dimensional homogeneity. For example, Newton's second law expresses the fact that

$$F = m \frac{d^2x}{dt^2},$$

where m is mass, x is a unit of length, and t is time. We can check, then, that the units do indeed match up on either side. Here, the RHS has units of $[m][x]/[t]^2$ or in SI units, kg m s^{-2} . This indeed matches our previous given SI unit decomposition for force.

Notice in addition that the input to functions like $\cos \theta$ and e^z must be non-dimensional (or dimensionless).

The process of nondimensionalisation is then as follows. Given an equation, we know that each term must have the same dimension. Therefore, we can scale all the dependent and independent variables by dimensional constants in order to yield a non-dimensional equation.

Why this is an important tool is demonstrated by the below.

3.3 Returning to the heat equation

Exact units are not relevant for dynamics, and it is instead the ratio of units that we care about. To apply this principle, let us non-dimensionalise the equation Equation 2.1. Let us consider the heat equation as given by the system

$$\begin{aligned}\frac{\partial T}{\partial t} &= \kappa \frac{\partial^2 T}{\partial x^2} \\ T(x, 0) &= T_{\text{init}} \\ T(0, t) &= T_a \\ T(L, t) &= T_b.\end{aligned}$$

Note that the constant κ is known as the thermal diffusivity, and is given by

$$\kappa = \frac{k}{\rho c},$$

which were the units previously introduced.

We introduce typical scales for each of the variables. For example, we non-dimensionalise the temperature, distance, and time by setting

$$T = [T]T', \quad x = [x]x', \quad t = [t]t'.$$

Remember that via the chain rule, we have that

$$\frac{d}{dx} = \frac{dx'}{dx} \frac{d}{dx'} = \frac{1}{[x]} \frac{d}{dx'}.$$

Substitution into the equation now yields

$$\begin{aligned}\frac{\partial T'}{\partial t'} &= \frac{\kappa[t]}{[x]^2} \frac{\partial^2 T}{\partial x^2} \\ T'(x', 0) &= \frac{T_{\text{init}}}{[T]} T \\ T'(0, t') &= \frac{T_a}{[T]} \\ T'(L/[x], t') &= \frac{T_b}{[T]}.\end{aligned}$$

Be sure to work out the above for yourself. For example, notice that the previous boundary at $x = L$ is now sent to $x' = L/[x]$. Now we see that the final equation is beautifully simple: At this point, we can identify a crucial non-dimensional parameter given by

$$\Pi = \frac{\kappa[t]}{[x]^2} = \frac{k[t]}{\rho c[x]^2}.$$

Although this looks quite complicated, you can test that this parameter is indeed non-dimensional by looking up the units for thermal conductivity, k , density ρ , and specific heat c , and verifying that indeed Π is non-dimensional.

3.4 Choice of units

It is important to remember that there are technically no wrong choices for the scales $[x]$, $[T]$, and $[t]$, as long as they yield consistent balances. However, some choices are better than others for the context of the problem. Let us select:

$$[x] = L \quad [T] = T_{\text{init}}.$$

This means our unit of length is “one pipe” (if this is indeed a pipe), and our unit of temperature is whatever the initial temperature was set to be. The choice of time is free at the moment, and therefore we can select time as

$$\Pi = \frac{\kappa[t]}{[x]^2} = 1 \implies [t] = \frac{L^2}{k}.$$

Now we see that the final equation is beautifully simple:

$$\begin{aligned} \frac{\partial T'}{\partial t'} &= \frac{\partial^2 T'}{\partial x'^2} \\ T'(x', 0) &= 1 \\ T'(0, t') &= \frac{T_a}{T_{\text{init}}} \equiv A, \\ T'(1, t') &= \frac{T_b}{T_{\text{init}}} \equiv B. \end{aligned} \tag{3.1}$$

3.5 Interpretation

What is the whole point of the above exercise?

In the original problem, there were five parameters:

$$\kappa, \quad T_{\text{init}}, \quad T_a, \quad T_b, \quad L.$$

Imagine a situation where you are performing an analysis of heat spread in different pipes of different materials (characterised by κ) and lengths (L), with different initial and boundary temperatures. That is potentially a highly complicated parameter space to search (five-dimensional).

Rather than solving the problem again and again for each change in parameters, what the analysis yielding (Equation 3.1) reveals is that the parameter space only needs to be two-dimensional, for parameters A and B . Therefore, by solving the problem at many different values of A and B , the solution space can be completely mapped.

Here is an example. Consider heat diffusion through common brick, which possesses a thermal diffusivity of

$$\kappa = 5.2 \times 10^{-7} \text{ m}^2/\text{s}.$$

Therefore a brick that is about $L = 10\text{cm} = 0.1\text{m}$ in size has a typical timescale of

$$[t] = \frac{L^2}{\kappa} \approx 1.9 \times 10^4 \text{ s} \approx 5 \text{ hrs.}$$

The above choice of $[t]$ is representative of the typical timescale you can expect for heat to diffuse through the brick. As you can see it's very long.

In summary, we have gone from a system where we needed to consider five parameters to a system where we only need to specify two (essentially, the ratio of the initial heat to the boundary values). This is an enormous simplification.

Moreover, your analysis has further identified the key non-dimensional parameters that appear, including:

$$\Pi = \frac{\kappa[t]}{[x]^2}, \quad A = \frac{T_a}{T_{\text{init}}}, \quad B = \frac{T_b}{T_{\text{init}}}, \quad \frac{L}{[x]}$$

In this particular problem, you had the freedom to choose L to remove one parameter (the ratio of lengths), and time could be chosen to set $\Pi = 1$.

3.6 GI Taylor and the energy of the atom bomb

During the third lecture, we covered the story of how non-dimensionalisation was famously used by British fluid dynamicist Geoffrey Ingram Taylor to estimate the energy of the atom bomb. The notes can be found in the scanned visualiser notes.

CHAPTER 4

Basic energy models

In this chapter, we introduce the simplest climate model to describe the temperature of the Earth. Despite its apparent simplicity, this basic climate model is the foundation of all further discussion. We will find that, according to the most basic model, the surface temperature of the Earth is expected to be much colder than it is actually; we conclude that the likely culprit is greenhouse gases.

Consider the Earth as a single averaged body. Our goal is to obtain an equation for the surface temperature, T , by considering the energy received from the sun versus the energy the Earth must emit into space. This kind of balance-equation procedure is used everywhere in mathematical modelling. In essence, we are looking to write down a precise statement of the following:

Basic energy balance

$$\text{Change in internal energy of Earth} = E_{\text{in}} - E_{\text{out}}, \quad (4.1)$$

where E_{in} is the incoming energy from the sun and E_{out} is the energy leaving the Earth. Note that the left hand-side (the internal energy) will depend on the temperature, T .

4.1 The basic energy model

The basic model is derived as follows.

1. By considering the incoming radiation from the sun, obtain an estimate for the incoming energy, E_{in} .
2. Use a constitutive law that indicates how much outgoing energy is produced by an object heated to some temperature.
3. Equate the change in internal energy to be equal to the difference between the above two items.

Note that the primary components of the global energy balance are radiative fluxes: we receive short-wave radiation (UV and visible light) from the Sun, and emit longwave radiation (infra-red) to space.

First we consider incoming energy.

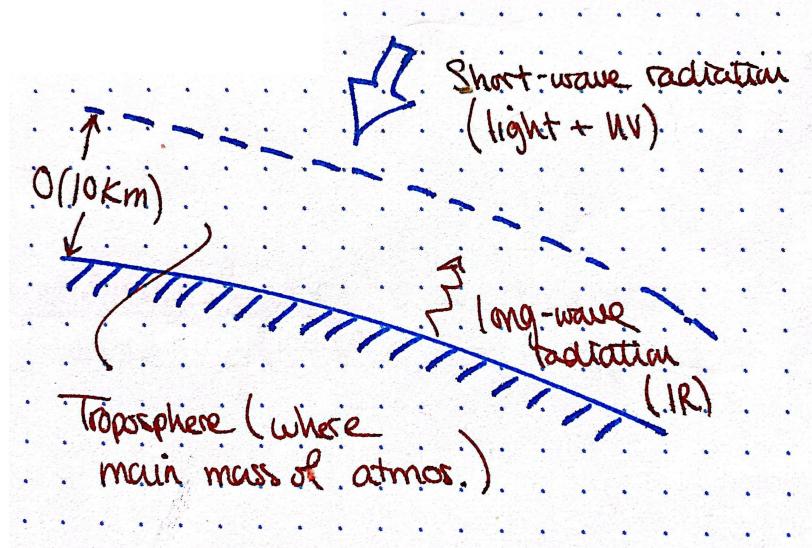


Figure 4.1: Radiation in the atmosphere

Energy from the Sun

First, note that the shortwave radiation (UV and radiation) received from the sun is $Q \approx 1370 \text{ W}\cdot\text{m}^{-2}$ (which we consider measured at a point near the planet). If we consider only that radiation that is absorbed into the Earth, we have

$$E_{\text{in}} = \pi R^2 Q (1 - a), \quad (4.2)$$

where R is the Earth's radius.

In the above formula, we have multiplied the flux, Q , with the visible surface area, πR^2 . There is an additional multiplication by $(1-a)$ where a is the *planetary albedo*, which characterises amount of energy reflected due to the surface properties. Light surfaces like snow will have high albedo, $a \approx 0.9$, while darker surfaces like the ocean have smaller albedo, $a \approx 0.3$. The global average albedo is $a \approx 0.3$.

Next we consider outgoing energy.

Energy from the Earth

We now wish to characterise the energy, E_{out} , and in the case of Earth, this will correspond to longwave radiation (infra-red) emitted into space. All bodies characterised by a temperature, say T_e , will emit radiation, Q_e . As a model, we can consider Q_e to be given by the **Stefan-Boltzmann law**, which states that

$$Q_e = \sigma T_e^4, \quad (4.3)$$

where $\sigma \approx 5.67 \times 10^{-8} \text{ W m}^{-2} \text{ K}^{-4}$ is the Stefan-Boltzmann constant.

Now although the Earth's surface may emit radiation according to (Equation 4.3), some of this radiation will be absorbed by the atmosphere and reflected back. This is the **greenhouse effect**. As a consequence of the greenhouse gas, the surface temperature of the Earth, T , will be larger than the effective emitting temperature, T_e . For the moment, we model this as

$$T_e = \gamma^{1/4} T, \quad (4.4)$$

where $\gamma < 1$ is a greenhouse gas factor, which depends on the properties of the atmosphere.

Now combining the above equations, we have

$$E_{\text{in}} - E_{\text{out}} = \pi R^2 Q(1 - a) - 4\pi R^2 \sigma \gamma T^4, \quad (4.5)$$

which gives the incoming energy per unit time.

Internal heat energy

Due to this incoming energy, the Earth will cool or heat in response. We need to know how the internal temperature of an object responds to an input in energy. The general formula is

$$\text{Internal heat energy} = \text{volume} \times (\rho c_p) \times T.$$

The key quantity is the experimentally determined, c_p , which corresponds to the **specific heat capacity**. It is given in the SI units of $\text{J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}$, i.e. energy per unit mass per unit temperature. Note that this applied to a shell around the planet of thickness d in the atmosphere, and so the mass is given by

$$\text{mass} = (4\pi R^2)d\rho \quad (4.6)$$

where ρ is the average density of the atmosphere. Let us imagine the increase in temperature, ΔT , during an interval of time, Δt . You can now verify that

$$[(4\pi R^2)d\rho]c_p(\Delta T)$$

returns the units of Joules—i.e. this is the internal energy produced during the time Δt . We then have

$$[(4\pi R^2)d\rho]c_p\Delta T = \Delta t(E_{\text{in}} - E_{\text{out}}).$$

Putting in (Equation 4.5) and taking the limit of $\Delta t \rightarrow 0$, we finally have a heat equation for the Earth's temperature.

Zero-dimensional model for the surface temperature of the Earth

The Earth's temperature, as measured on a layer of thickness d at the troposphere is given by the following ordinary differential equation (ODE) for $T = T(t)$:

$$C \frac{dT}{dt} = \frac{1}{4}Q(1 - a) - \sigma \gamma T^4, \quad (4.7)$$

where we have defined

$$C = \rho c_p d$$

as the heat capacity of the atmosphere. Above, the solar flux, Q is often taken to be $Q = 1370 \text{ W/m}^2$, $\sigma = 5.67 \times 10^{-8} \text{ W/(m}^2\text{K}^4)$, and $\gamma \leq 1$ is the greenhouse gas factor.

The above equation Equation 4.7 is time-dependent, but we may consider that the surface temperature, either over long-time or in an averaged sense, is dictated by the steady-

state (time-independent) solution. Setting dT/dt to zero, we see that there is a unique steady-state given by

$$T = \left(\frac{Q(1-a)}{4\sigma\gamma} \right)^{1/4}.$$

If we take $Q \approx 1370 \text{ W m}^{-2}$, $a \approx 0.3$, $\sigma \approx 5.67 \times 10^{-8} \text{ W m}^{-2} \text{ K}^{-4}$, we then get

$$T \approx 255 \text{ K} = -18^\circ \text{C}.$$

under the assumption that $\gamma = 1$. That's pretty cold! The actual average temperature is around $288 \text{ K} \approx 15^\circ \text{C}$.

The above back-of-the-envelope calculation seems to suggest that the parameter $\gamma < 1$ plays an important role in keeping the Earth warm enough for us to live on, and indeed the value of γ inferred by the above is roughly $\gamma \approx 0.61$. Later on in the course, we will develop a more rigorous model to predict such a γ by studying the properties of the atmosphere.

4.2 The history of global warming

The history of global warming (and hence the estimation of γ) is convoluted, but the origins can be considered as far back as the work of (Fourier, 1827) and (Pouillet, 1838), and is discussed in the work by (Van der Veen, 2000).

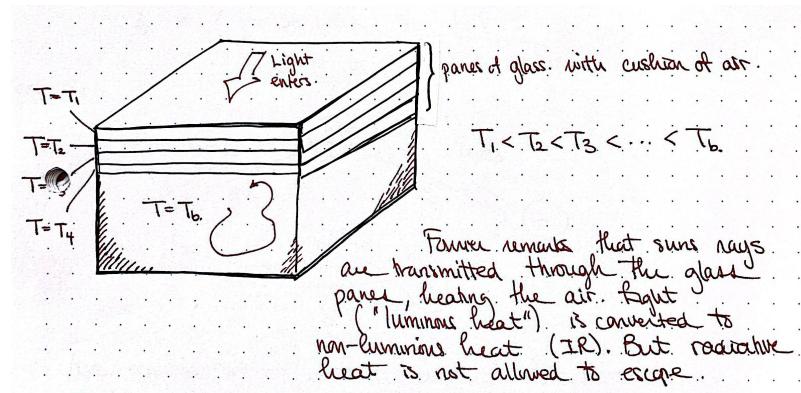


Figure 4.2: An illustration of Fourier's glass box

4.3 Next in the study of energy balance models?

At this stage, the basic energy balance model studied in this chapter can be made more complex and realistic through many different extensions. For instance:

- Consider the effects of a non-constant and nonlinear albedo. Since the albedo depends on the material property (water vs. ice vs. land), we can incorporate a toy model for spatial variability by allowing the albedo to depend on temperature, $a = a(T)$.
- Consider the time-dependence. What happens if the system starts out of the steady state? Does it tend to relax to its equilibrium profile? Can we analyse the temporal properties of the solutions?
- Can we incorporate latitude (or longitudinal) dependence into the model. For instance, should the albedo and solar radiation be considered as functions of the latitude?

- Should we incorporate the effects of diffusion (heat spreading out) or convection (heat transported with flows)?
- To what extent should we consider the effects of the atmosphere on the absorption and reflection properties of the incoming (and outgoing) energy?

Many of these questions are much more involve, as they require differential or partial differential equations. Therefore in the next few chapters, we will discuss, in a very general way, the collection of applied and numerical mathematics techniques that can be used when studying problems in the physical sciences.

Part II

Practical applied mathematics

As we go deeper into formulating the equations that model or govern aspects of Planet Earth, we will quickly come to the realisation that many such equations, even for the simplest minimal models, are not exactly solvable.

For example, in the analysis of the Ocean models later in the course, we develop the following “simple” model for the temperature in the ocean:

$$\begin{aligned}\frac{dx}{dt} &= \delta(1 - x) - |f(x, y)|x, \\ \frac{dy}{dt} &= 1 - y - |f(x, y)|y,\end{aligned}$$

where we have introduced the function,

$$f(x, y; R, \lambda) = \frac{1}{\lambda}(Rx - y),$$

This is quite a difficult problem! This is essentially a set of two nonlinear differential equations for two unknowns and three parameters. What kind of practical applied mathematics can we apply to study such problems?

The intention of this part is to introduce (and in some cases, review) three key concepts:

1. Asymptotic approximations.
2. Numerical solutions of differential equations.
3. Numerical solutions of nonlinear equations (Newton’s method).

as well as any additional tools you may need to study the kinds of differential equations involved in Maths of Planet Earth!

CHAPTER 5

Asymptotic approximations I

One powerful set of techniques for approximating solutions to equations is called *asymptotic analysis* or *perturbation theory*. To begin with, in this chapter, we introduce you to these techniques as a means to approximating the solutions to equations like:

$$\epsilon x^2 + x - 1 = 0,$$

when ϵ is a small parameter.

Soon, we apply the techniques to approximating solutions of differential equations.

5.1 A simple quadratic

A singular quadratic

Consider the solution of

$$\epsilon x^2 + x - 1 = 0, \quad (5.1)$$

where ϵ is a fixed and very small positive number, say 0.000001. Forget that we know how to solve a quadratic equation: is it possible to develop a systematic approximation method?

If $\epsilon = 0$, then $x = 1$. Moreover, if we substitute $x = 1$ into the equation, then we see that the error is small and proportional to ϵ . It is natural to seek an approximation in powers of ϵ . We call this an *asymptotic expansion*. We write

$$x = x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots$$

Substitution into the equation yields

$$\epsilon \left(x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots \right)^2 + \left(x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots \right) - 1 = 0.$$

Expand and collect terms in powers of ϵ :

$$\left(x_0 - 1 \right) + \epsilon \left(x_1 + x_0^2 \right) + \epsilon^2 \left(x_2 + 2x_0 x_1 \right) + \dots = 0.$$

Now we equate coefficients at each order in ϵ . This gives

$$x_0 - 1 = 0 \implies x_0 = 1$$

$$x_1 + x_0^2 = 0 \implies x_1 = -1$$

$$x_2 + 2x_0 x_1 = 0 \implies x_2 = 2$$

We therefore have obtained the three-term approximation,

$$x = 1 - \epsilon + 2\epsilon^2 + \dots$$

Clearly we could continue this process *ad infinitum* obtaining increasingly accurate approximations to one of the roots.

The singular root

But where has the other quadratic root gone?

The problem is that in considering ϵ to be small, we began by ignoring the leading term, ϵx^2 . We effectively assumed that the equation was primarily *balanced* by setting the x term with the -1 term, and the sum of the two terms approximately equalling zero.

But if $|x|$ is large, then clearly our assumption that ϵx^2 being small may not be necessarily true for it depends on how large $|x|$ is compared to ϵ . Note that if $|x|$ is large, then necessarily the last term, -1 , is negligible in comparison. Therefore, in order for ϵx^2 to balance x , we see that $|x|$ must be of size $1/\epsilon$.

Therefore this suggests that we should re-scale our solution as follows

$$x = \frac{X}{\epsilon}.$$

Substitution into the original quadratic now yields

$$X^2 + X - \epsilon = 0.$$

Now notice that $\epsilon = 0$ expresses the correct balance in order to detect that missing root. Again we write

$$X = X_0 + \epsilon X_1 + \epsilon^2 X_2 + \dots$$

and attempt to solve order by order. Substitution into the equation yields

$$\left(X_0 + \epsilon X_1 + \epsilon^2 X_2 + \dots\right)^2 + \left(X_0 + \epsilon X_1 + \epsilon^2 X_2 + \dots\right) - \epsilon = 0.$$

Expand and collect orders of ϵ :

$$\begin{aligned} X_0^2 + X_0 &= 0 \implies X_0 = -1 \\ 2X_0 X_1 + X_1 - 1 &= 0 \implies X_1 = -1, \end{aligned}$$

and thus to two orders, we have

$$X = -1 - \epsilon + \dots \implies x = -\frac{1}{\epsilon} - 1 + \dots$$

Of course, we have used a very simple example (a solvable quadratic) to illustrate the idea of asymptotic approximations, but you should hopefully see that this method is extensible to much more complicated equations.

```
import numpy as np
import matplotlib.pyplot as plt

ep = np.linspace(1, 0.1, 20)
root1 = (-1 + np.sqrt(1-4*ep*(-1)))/(2*ep)
```

```

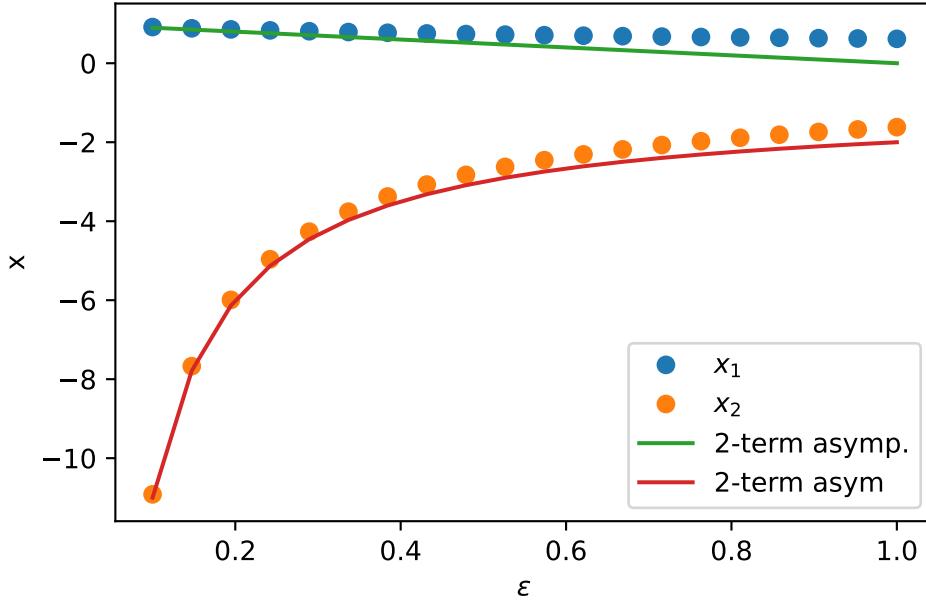
root2 = (-1 - np.sqrt(1-4*ep*(-1)))/(2*ep)
asym1 = 1 - ep
asym2 = -1/ep - 1
plt.plot(ep, root1, 'o')
plt.plot(ep, root2, 'o')
plt.plot(ep, asym1, '-')
plt.plot(ep, asym2, '-')
plt.legend(['$x_1$', '$x_2$', '2-term asymp.', '2-term asym'])
plt.xlabel('$\epsilon$')
plt.ylabel('x');

```

<>>:14: SyntaxWarning: invalid escape sequence '\e'

<>>:14: SyntaxWarning: invalid escape sequence '\e'

/var/folders/lw/7v5bfgnj6gj4qhr6fykhcp3h0000gn/T/ipykernel_87864/3662636573.py:14: Sy
plt.xlabel('\$\epsilon\$')



5.2 Order notation and the tilde sign for asymptotic

We define precisely what we mean when we say that two functions, say f and g , exhibit the same behaviour in some limit, say $\epsilon \rightarrow 0$ or $x \rightarrow x_0$ or $x \rightarrow \infty$ and so forth. For instance, we claim that the graphs of $\sin(x)$ and x look very similar as $x \rightarrow 0$. Thus we might write

$$\sin(x) \sim x \quad \text{as } x \rightarrow 0. \quad (5.2)$$

This notation of similarity allows us to specify functional behaviours at a deeper level than just limits. As you can see, it is not as useful to specify that

$$\lim_{x \rightarrow 0} \sin x = \lim_{x \rightarrow 0} x.$$

In contrast, the asymptotic relation is much more prescriptive about the way that the functions are approaching the limit.

Definition of \sim , \gg , and \ll

First, the notation

$$f(x) \ll g(x), \quad x \rightarrow x_0,$$

is read as “ $f(x)$ is much smaller than $g(x)$ as $x \rightarrow x_0$ ” and means

$$\lim_{x \rightarrow x_0} \frac{f(x)}{g(x)} = 0.$$

We may analogously use $g(x) \gg f(x)$ for “much greater than...”.

Second, the notation

$$f(x) \sim g(x), \quad x \rightarrow x_0,$$

is read as “ $f(x)$ is asymptotic to $g(x)$ as $x \rightarrow x_0$ ”, and means that the error between f and g tends to zero as $x \rightarrow x_0$, or

$$\lim_{x \rightarrow x_0} \frac{f(x)}{g(x)} = 1.$$

We will often say “ f is like g ” or “ f behaves like g ”,

Here are some examples.

Examples

- $\sin x \sim x \sim \tan x$ as $x \rightarrow 0$
- $x^2 + x + 1 \sim \frac{x^3 + \sin x}{1+x}$ as $x \rightarrow \infty$
- $\sin x \ll \cos x$ as $x \rightarrow 0$

In the examination of limiting processes, often the main issue of consideration is the relative sizes of quantities defined according to their powers. For example, if x is a very small number, with $x = 10^{-5}$, then x^5 is much smaller than x (in terms of our notation, $x^5 \ll x$ as $x \rightarrow 0$). On the other hand, we might not care so much about the difference between

$$x^5 \quad \text{vs.} \quad 5x^5$$

The point is that the *order* of x^5 and $5x^5$ is the same as $x \rightarrow 0$. The “Big-Oh” notation formalises this distinction.

Definition of Big-Oh

We write $f = O(g)$ as $x \rightarrow x_0$ to mean that there exists constants $K > 0$ and $x^* > 0$ such that

$$|f| < K|g| \quad \text{for all } |x - x_0| < x^*.$$

In practice, the use of the order symbol is very natural and you will not need to work with the technical definition. For example, when you derive the terms of the Maclaurin/Taylor series, you are naturally clustering all the terms of the same order (power) together. For us, the O symbol provides a very convenient way of separating terms of different sizes.

Examples

- $2 \sin x = O(\tan x)$ as $x \rightarrow 0$
- $x^2 + x + 1 = O\left(\frac{5x^3 + \sin x}{1+x}\right)$ as $x \rightarrow \infty$

Let us return to the case of the quadratic example (Equation 5.1). Using the O notation, we can write

$$x = \begin{cases} 1 - \epsilon + 2\epsilon^2 + O(\epsilon^3) \\ -\frac{1}{\epsilon} - 1 + O(\epsilon^2) \end{cases}$$

for the two roots. Alternatively, we can truncate the expansions and simply using the \sim symbol:

$$x \sim \begin{cases} 1 - \epsilon \\ -\frac{1}{\epsilon} - 1 \end{cases}$$

CHAPTER 6

Asymptotic approximations II

In the previous chapter, we learned about the technique of asymptotic expansions, whereby the solution of an equation is expressed in terms of a series in powers of a small parameter:

$$x = x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots$$

The precise choice of power progression (here integer powers of ϵ will depend on the particular problem. The same idea can be extended to approximating solutions of differential equations. The upshot of this procedure is that at each order of the scheme, a simpler problem can be studied.

Again it is best to demonstrate through examples.

6.1 Returning to the projectile problem

In Chapter 40 and Chapter 25 you studied the non-dimensionalisation of the projectile problem. Once re-scaled, it takes the following form:

$$\begin{aligned} \frac{d^2y}{dt^2} &= -\frac{1}{(1 + \epsilon y)^2}, & t > 0 \\ y(0) &= 0, \\ y'(0) &= 1. \end{aligned} \tag{6.1}$$

This is a difficult problem without, in fact, any explicit solutions. However, we can estimate the solution in the limit $\epsilon \rightarrow 0$. We expand the solution as

$$y(t) = y_0(t) + \epsilon y_1(t) + \epsilon^2 y_2(t) + \dots$$

In order to expand the denominator, you can use Taylor's theorem to expand the function

$$f(x) = (1 + x)^\alpha = f(0) + f'(0)x + \dots = 1 + \alpha x + \dots$$

around $x = 0$.

The differential equation now yields

$$y_0'' + \epsilon y_1'' + \epsilon^2 y_2'' + \dots = -[1 - 2\epsilon(y_0 + \epsilon y_1 + \dots) + \dots]$$

so grouping terms together order-by-order yields

$$[y_0'' + 1] + \epsilon [y_1'' - 2y_0] + \dots = 0.$$

We can similarly substitute the expansion into the initial conditions. Altogether, at leading order, we obtain the following system to solve:

$$\begin{aligned} y_0'' + 1 &= 0, \\ y_0(0) &= 0, \\ y_0'(0) &= 1. \end{aligned}$$

Integrating twice and applying the boundary conditions gives us

$$y_0(t) = -\frac{1}{2}t^2 + t.$$

In fact, this is simply the parabolic motion you would expect from school Physics. The $\epsilon = 0$ solution corresponds to assuming that the mass at the centre of the planet is dominant and then acceleration is constant.

However, we can now proceed to higher order and examine the nonlinear effects. Proceeding to $O(\epsilon)$, we have the following system to solve:

$$\begin{aligned} y_1'' &= 2y_0, \\ y_1(0) &= 0, \\ y_1'(0) &= 0. \end{aligned}$$

Notice the boundary conditions come from the fact there are no ϵ corrections in the original boundary conditions, so $y_n(0) = y_n'(0) = 0$ for all $n > 0$. Again this system is simple to integrate. Integrating the solution for y_0 twice and substitution of the initial conditions yields

$$y_1(t) = -\frac{1}{12}t^4 + \frac{1}{3}t^3.$$

We have thus solved for the asymptotic approximation to two orders. We have

$$y(t) \sim \left[-\frac{1}{2}t^2 + t \right] + \epsilon \left[-\frac{1}{12}t^4 + \frac{1}{3}t^3 \right].$$

This was quite an accomplishment! We have taken a problem that was not easily solvable in explicit form and through fairly simple integrations, obtained an approximation to two orders in ϵ . How good is it? Let us solve the problem numerically and compare with the asymptotic approximation.

6.2 Numerical solutions of IVPs

We first demonstrate how to solve ODEs (initial-value-problems, IVPs) using black-box functions in Python. For starters, most numerical formulations for ODEs will require that the problem be posed in terms of a first-order system of equations. To convert (Equation 6.1) into such a form, create a set of unknowns for the derivatives. Set

$$\mathbf{Y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} = \begin{pmatrix} y(t) \\ y'(t) \end{pmatrix}$$

Then we have the following first-order system:

$$\begin{aligned}\mathbf{Y}'(t) &= \mathbf{F}(t, \mathbf{Y}(t)) = \begin{pmatrix} y'_1 \\ -\frac{1}{(1+\epsilon y_1)^2} \end{pmatrix} \\ \mathbf{Y}(0) &= \begin{pmatrix} 0 \\ 1 \end{pmatrix}\end{aligned}\tag{6.2}$$

You can find a little guide on using `solve_ivp` in Python [here](#). Here is the Python code to solve the differential equation.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp

ep = 0.2 # epsilon value
tmax = 2 # max time
t = np.linspace(0, tmax, 100) # mesh used for plotting

# Define function for the ODE
def f(t, Y):
    ep = 0.2
    y, yp = Y
    ypp = -1/(1 + ep*y)**2
    return [yp, ypp]

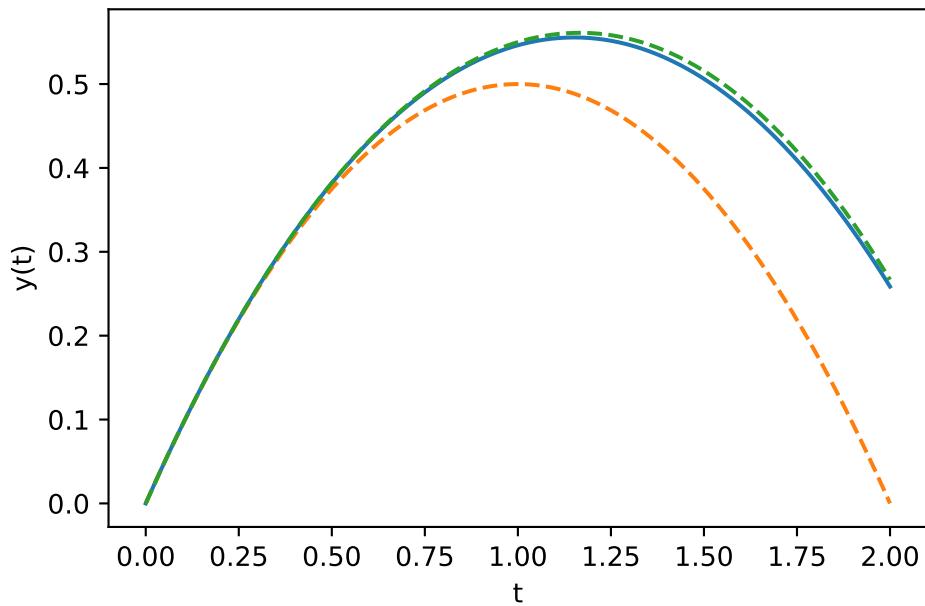
# define the initial condition
Y0 = [0, 1]

sol = solve_ivp(f, [0, tmax], Y0, dense_output=True)

# Prior to plotting, re-interpolate solution on a fine grid
yy = sol.sol(t)
# Asymptotic solutions
y0 = -1/2*t**2 + t
y1 = -1/12*t**4 + 1/3*t**3

# Plot it all
plt.plot(t, yy[0,:])
plt.plot(t, y0, '--')
plt.plot(t, y0 + ep*y1, '--')
plt.xlabel('t')
plt.ylabel('y(t)')

Text(0, 0.5, 'y(t)')
```



The two-term approximation does beautifully well, even at this moderate value of $\epsilon = 0.2$.

7

CHAPTER

Numerical solutions of IVPs

In the previous section, we used built-in ODE solvers to develop numerical solutions. It is important to gain an understanding how a simple ODE solver works. The simplest scheme is called [Euler's method](#), and this we now explain.

Begin from the system (Equation 6.2). We assume that the solution is represented by a discrete set of points, $\mathbf{Y}_n = \mathbf{Y}(t_n)$ at the times $t_0 = 0$, $t_1 = \Delta t$, $t_2 = 2\Delta t$, and so on. The time derivative is written as a discrete derivative while we approximate the right hand side by its value at the nth time step:

$$\frac{\mathbf{Y}_{n+1} - \mathbf{Y}_n}{\Delta t} = \mathbf{F}(t_n, \mathbf{Y}_n)$$

Rearranging yields a very simple algorithm for solving the ODE:

$$\mathbf{Y}_n = \mathbf{Y}_{n-1} + \mathbf{F}(t_{n-1}, \mathbf{Y}_{n-1})\Delta t$$

for $n = 1, 2, 3, \dots$

This would be implemented via the following pseudocode:

Euler's method

1. Input: function $f(t, Y)$
time step, dt
initial condition, Y_0
2. Set initial condition $Y = Y_0$
2. Take one Euler step and overwrite previous value
$$Y = Y + f(t, Y)$$
3. Increment t by dt and goto 2

Euler's method is conceptually simple but quite inaccurate. But in this case, we see that it works fairly well in comparison to the built-in solvers.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp

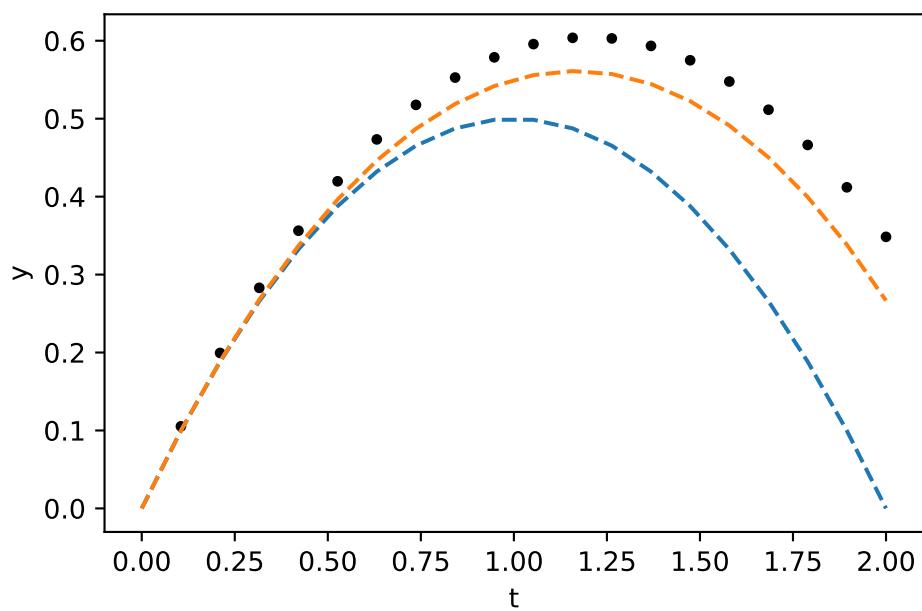
ep = 0.2          # epsilon value
tmax = 2          # max time
N = 20           # number of steps
t = np.linspace(0, tmax, N) # mesh used for plotting
dt = t[1] - t[0]

# Define function for the ODE
def f(t, Y, ep):
    y, yp = Y
    ypp = -1/(1 + ep*y)**2
    return np.array([yp, ypp])

# define the initial condition
Y = [0.0, 1.0]
ti = 0

# define the solution vector
for i in range(1, N):
    ti = ti + dt # Increment time
    Y = Y + f(ti, Y, ep)*dt # Euler step
    plt.plot(ti, Y[0], 'k.')

# Asymptotic solutions
y0 = -1/2*t**2 + t
y1 = -1/12*t**4 + 1/3*t**3
plt.plot(t, y0, '--')
plt.plot(t, y0 + ep*y1, '--')
plt.xlabel('t');
plt.ylabel('y');
```



CHAPTER 8

Asymptotic approximations III

8.1 Regular vs. singular problems

In Chapter 6, we studied how asymptotic expansions can be used to approximate equations like

$$\frac{d^2y}{dt^2} = -\frac{1}{(1+\epsilon y)^2}$$

by expanding the solution as $y(t) = y_0(t) + \epsilon y_1(t) + \dots$. These are known as *regular* problems because a small perturbation, ϵ , does not seem to fundamentally change the $\epsilon = 0$ solution beyond a small perturbation. This is not always the case. In *singular* problems, the situation of $\epsilon \neq 0$ is fundamentally different than the situation from $\epsilon = 0$. You have already seen such an example in Chapter 5. The equation

$$\epsilon x^2 + x - 1 = 0$$

has one root for $\epsilon = 0$ and two roots for non-zero small ϵ —even infinitesimally small values! This is quite interesting. From a wider scientific perspective, you may wonder what other problems in nature possess such singular effects.

The point of this lecture is study a technique known as *matched asymptotics*. These matched asymptotics are often necessary for singularly perturbed differential equations.

8.2 A singular first-order ODE problem

Previously in Chapter 4, we derived a basic equation that governs the temperature on the surface of the planet. This equation had the following form:

$$(\rho c_p V) \frac{dT}{dt} = E_{\text{in}}(t, T) - E_{\text{out}}(t, T).$$

For the purpose of this section, let us make up a toy model. We suggest, in non-dimensional form,

$$\begin{aligned} \epsilon \frac{dT}{dt} &= R(t) - T, \quad t \geq 0 \\ T(0) &= T^*, \end{aligned} \tag{8.1}$$

where we consider $\epsilon > 0$ and $\epsilon \ll 1$. You can think of the above model as modelling the temperature on a substance that radiates heat in a fashion proportional to itself ($-T$)

and is being subjected to an (incoming) heat source, R . Let us take as an example,

$$R(t) = 1 + A \cos(t).$$

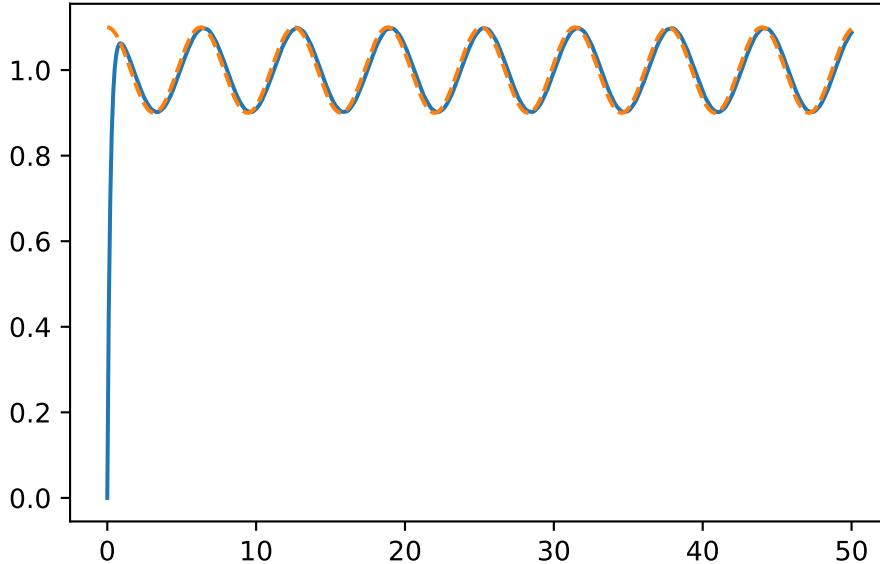
Our choice for R is not so important. This equation is, in fact, solvable in closed form (how?) but let us get additional practice solving numerically.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp

Ts = 0
ep = 0.2
A = 0.1
f = lambda t, T: 1/ep*(1 + A*np.cos(t) - T)

tmax = 50
sol = solve_ivp(f, [0, tmax], [Ts], dense_output=True)

t = np.linspace(0, tmax, 1000)
y = sol.sol(t)
plt.plot(t, y[0])
plt.plot(t, 1 + A*np.cos(t), '--')
```



What do we observe? If $\epsilon = 0$, then we expect the solution $T \sim R(t)$. This is shown with the dashed line. However, this solution does not satisfy the necessary initial condition. We observe that near $t = 0$, the exact solution seems to very rapidly diverge from the approximation in order to satisfy the proper boundary condition. The region in t where this rapid change occurs is called a *boundary layer*.

If we repeat the experiment with an even smaller value of ϵ , we would observe that the size of this boundary layer seems to tends to zero as $\epsilon \rightarrow 0$. This numerical experiment thus inspires the following method.

8.3 Boundary layer theory

We seek a method that will allow us to develop a uniformly valid approximation, i.e. an approximation that is good everywhere in the relevant domain, $t \geq 0$. Begin by performing the usual asymptotic approximation:

$$T(t) = T_0(t) + \epsilon T_1(t) + \epsilon^2 T_2(t) + \dots$$

Substitution into the ODE (Equation 8.1) yields at leading order,

$$0 = R(t) - T_0(t) \implies T_0(t) = R(t) = 1 + A \cos(t).$$

As we have noted, this approximation fails to satisfy the initial condition, $T(0) = T^*$ in general. It is possible to go to higher order but this is not so important at the moment. So for now, we have obtained:

$$T_{\text{outer}} \sim [1 + A \cos t].$$

We have chosen to refer to this as the *outer* solution for reasons that will be abundantly clear. But rather than satisfying $T(0) = T^*$, this approximation has the limiting behaviour of

$$\lim_{t \rightarrow 0} T_{\text{outer}} \sim [1 + A]$$

Above, we have only included the leading term in the limit expression.

The inner scaling

Our intuition follows a very similar logic to the examination of the singular root in Section 5.1. Above, our naive assumption was that $\epsilon T'(t)$ could be ignored since ϵ is a small number. However, this may not be the case if the gradient is very large.

Our intuition further suggests that the boundary layer occurs near $t = 0$ and that it scales in size with ϵ . Therefore, let us set

$$t = \epsilon^\alpha s,$$

as a change of coordinates. We expect $\alpha > 0$ (otherwise t is not small), and within this region, we expect the new coordinate, s , to be $O(1)$ (of moderate size). We then transform the unknown function:

$$T(t) = T(\epsilon s) = U(s),$$

and seek a new differential equation for U . By the chain rule,

$$\frac{dT}{dt} = \epsilon^{-\alpha} \frac{dU}{ds}.$$

Before substituting into the equation, we are prudent to examine the behaviour of $R(t)$ near $t = 0$. We know by Taylor's theorem that

$$R(t) = 1 + A \left(1 - \frac{t^2}{2} + \dots \right).$$

Therefore, under the substitution, we may approximate R by its leading terms:

$$R(\epsilon^\alpha s) \sim 1 + A.$$

For now, we will not need more terms than this. Substituting into the ODE now gives

$$\epsilon^{1-\alpha} \frac{dU}{ds} \sim [1 + A] - U.$$

Now in order to involve the first term, it is sensible to select

$$1 - \alpha = 0 \implies \alpha = 1.$$

The inner equation

Therefore, the correct coordinate re-scaling was the ‘obvious’ one:

$$t = \epsilon s.$$

Substituting this again in (Equation 8.1):

$$\begin{aligned}\frac{dU}{ds} &= 1 + A \cos(\epsilon s) - U, \\ U(0) &= T^*.\end{aligned}$$

The procedure is now exactly the same. We expand

$$U(s) = U_0(s) + \epsilon U_1(s) + \epsilon^2 U_2(s) + \dots$$

At leading order, we get

$$\begin{aligned}U'_0 &= 1 + A - U_0 \\ U_0(0) &= T^*.\end{aligned}$$

The above ODE can be solved by integrating factors. Multiplying both sides by e^s , we have

$$(U_0 e^s)' = (1 + A)e^s.$$

Integrate and use the initial condition:

$$U_0(s) = (1 + A) + (T^* - (1 + A))e^{-s}.$$

This is exactly what we expect. Notice that

$$\lim_{s \rightarrow \infty} U_0(s) = \lim_{t \rightarrow 0} T_{\text{outer}} \quad (8.2)$$

therefore the outer limit of our inner solution matches the inner limit of our outer solution. In terms of outer coordinates, our inner solution is approximated as follows:

$$T_{\text{inner}} \sim (1 + A) + (T^* - (1 + A))e^{-t/\epsilon}.$$

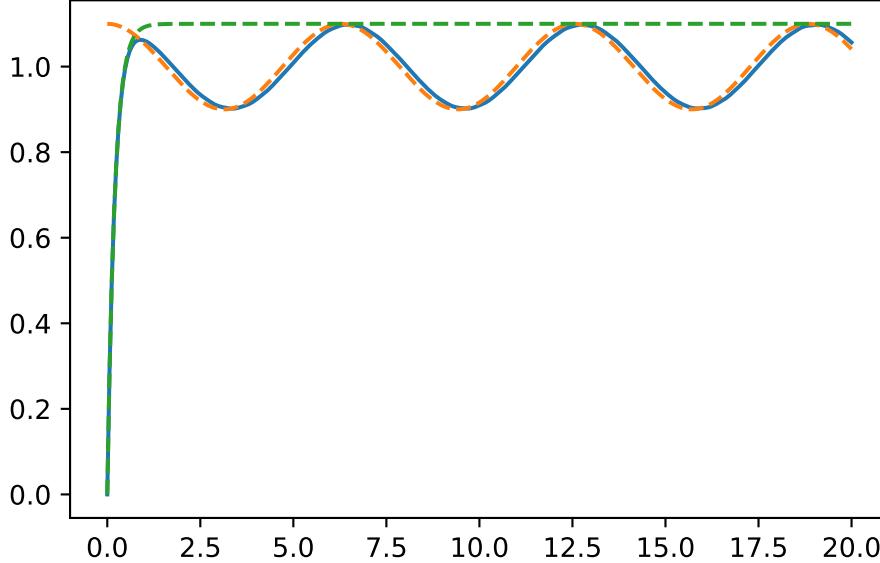
Let’s finally plot this with our previous curves:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp

Ts = 0
ep = 0.2
A = 0.1
f = lambda t, T: 1/ep*(1 + A*np.cos(t) - T)

tmax = 20
sol = solve_ivp(f, [0, tmax], [Ts], dense_output=True)

t = np.linspace(0, tmax, 1000)
y = sol.sol(t)
plt.plot(t, y[0])
plt.plot(t, 1 + A*np.cos(t), '--')
plt.plot(t, (1 + A) + (Ts - (1 + A))*np.exp(-t/ep), '--')
```



It works beautifully!

Summary

Let us summarise the procedure of matched asymptotics.

1. Expand the solution of the differential equation naively in the typical asymptotic expansion (e.g. in powers of ϵ).
2. Notice that the approximation does not satisfy certain boundary conditions.
3. Re-scale the coordinates in the ‘inner’ regions.
4. Develop an inner solution that satisfies the boundary condition. Ensure it matches the outer solution.

You will get more practice of this procedure in the problem sets.

An additional problem

In Lecture 8 (26 Feb 2025), problem class, and associated problem set, we studied the example of

$$\epsilon T'' + 2T' + T = 0, \quad t \in [0, 1], \quad (8.3)$$

$$T(0) = 0, \quad T(1) = 1. \quad (8.4)$$

This is an example where the inner solution, near $t = 0$, requires the solution of a second-order differential equation that requires matching to the outer solution. You can follow the matching procedure via the visualiser notes (and it is further touched-upon in the problem set).

CHAPTER 9

Nonlinear root finding

Before we move on back to the subject of applications, it is worth providing an introduction to methods for solving nonlinear equations. Many problems you will encounter in applications, including for ordinary or partial differential equations, can be re-formulated as the solution of a nonlinear system of equations.

Newton's method is the most well-known scheme for solving nonlinear equations. Suppose we wish to solve the scalar equation,

$$f(x) = 0,$$

given some initial guess, $x = x_0$, of the root. Suppose the root lies at $x = x^*$. Then by Taylor's theorem,

$$f(x^*) = f(x_0) + f'(x_0)(x^* - x_0) + O(f''(x_0)(x - x_0)^2).$$

If we assume the quadratic terms are negligible then solve for x^* this gives

$$x^* \approx x_0 - \frac{f(x_0)}{f'(x_0)}.$$

There is a geometrical interpretation of the above. Essentially, in order to estimate the root of $f(x) = 0$, we have used the tangent line at the point $x = x_0$, and used the intersection of this tangent line with the axis as the approximation. This procedure can then be iterated.

Thus, provided that the desired root, x^* , is such that $f'(x^*) \neq 0$, and x_0 is sufficiently close to x^* , then the following iterates converge to the root:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

9.1 Demo of Newton's method for scalar equations

Here is a simple demonstration of Newton's method in order to solve for one of the roots of the following:

$$f(x) = x^3 + x - 1.$$

We will start with the initial guess of $x_0 = 0$. We can do this by hand in lectures using a pocket calculator. The solution is $x^* \approx 0.6823278..$

i	xi	f(xi)	f'(xi)	-f(xi)/f'(xi)	error
0					
1					
2					

```
import numpy as np

x0 = 0
N = 10

def Newton(f, df, x, maxiter=10):
    i = 0
    while (i < maxiter):
        err = f(x)
        x = x - err / df(x)
        print("f(x) = ", np.abs(err), " , x = ", x)
        i = i + 1
    return x, err

f = lambda x: x**3 + x - 1
df = lambda x: 3*x**2 + 1

x, err = Newton(f, df, x0, N)
print("Final approximation = ", x)
```

```
f(x) = 1 , x = 1.0
f(x) = 1.0 , x = 0.75
f(x) = 0.171875 , x = 0.686046511627907
f(x) = 0.008941036638283384 , x = 0.6823395825973142
f(x) = 2.823062168566537e-05 , x = 0.6823278039465127
f(x) = 2.839946056099052e-10 , x = 0.6823278038280194
f(x) = 2.220446049250313e-16 , x = 0.6823278038280193
f(x) = 1.1102230246251565e-16 , x = 0.6823278038280193
f(x) = 1.1102230246251565e-16 , x = 0.6823278038280193
f(x) = 1.1102230246251565e-16 , x = 0.6823278038280193
Final approximation = 0.6823278038280193
```

It is a good idea to also learn how to do this using built-in packages. The ‘fsolve’ function provides a Newton-like nonlinear solver. In fact, it can estimate the Jacobian (derivative), so only the function values need to be provided.

```
import numpy as np
from scipy.optimize import fsolve

f = lambda x: x**3 + x - 1

x0 = 1
x, info, ier, msg = fsolve(f, x0, full_output=True)
```

```
print(msg)
print(x)
```

The solution converged.
[0.6823278]

9.2 Newton's method for systems of nonlinear equations

Newton's method generalises naturally to the case of a system of equations. Suppose we wish to solve for the n unknowns $\mathbf{x} = (x_1, \dots, x_n)$ via

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} F_1(\mathbf{x}) \\ F_2(\mathbf{x}) \\ \vdots \\ F_n(\mathbf{x}) \end{pmatrix} = 0.$$

We have, via Taylor's formula,

$$\mathbf{F}(\mathbf{x}_{i+1}) \sim \mathbf{F}(\mathbf{x}_i) + J(\mathbf{x}_i)(\mathbf{x}_{i+1} - \mathbf{x}_i) + \mathcal{O}(||\mathbf{x}_{i+1} - \mathbf{x}_i||^2),$$

where J is the Jacobian matrix

$$J(\mathbf{x}) = \nabla \mathbf{F}(\mathbf{x}) = \begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \cdots & \frac{\partial F_n}{\partial x_n} \end{pmatrix}.$$

Therefore, Newton's method forms the iterates of

$$\mathbf{x}_{i+1} = \mathbf{x}_i - J^{-1}(\mathbf{x}_i)\mathbf{F}(\mathbf{x}_i),$$

which takes a very similar form to the scalar case.

However, solution of the inverse of J is typically inefficient, and it is better to instead solve for $\delta_{i+1} = \mathbf{x}_{i+1} - \mathbf{x}_i$ via

$$J(\mathbf{x}_i)\delta_{i+1} = -\mathbf{F}(\mathbf{x}_i),$$

and then calculate $\mathbf{x}_{i+1} = \mathbf{x}_i + \delta_{i+1}$. There are many ways of solving the above matrix problem efficiently using built-in routines that perform, e.g. Gaussian elimination.

9.3 Secant method

In many situations, evaluation of the Jacobian (or derivative) is the most time-consuming or difficult part of a nonlinear solver. Built-in solvers like 'fsolve', in fact, have the ability to approximate the derivative numerically.

The Secant Method is similar to Newton's Method but replaces the derivative by a finite difference. Geometrically, the tangent line is replaced with a line through the two last known guesses. The algorithm goes as follows.

Secant method

1. Develop two initial guesses to the solution, x_0 and x_1
2. Compute

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

CHAPTER 10

Numerical solutions of PDEs

We shall begin by introducing the simplest finite-difference routine for solving a PDE using Euler time stepping. Let us consider the solution of the heat equation on a finite interval:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \\ u(0, t) = 1, \quad u(1, t) = 2, \quad u(x, 0) = 2.$$

10.1 Finite difference problem

We consider a discretisation of the spatial domain $x \in [0, 1]$ via the $(N + 1)$ points

$$x_0 = 0, \quad x_1 = \Delta x, \quad x_2 = 2\Delta x, \quad \dots, \quad x_{N+1} = 1.$$

Typically Δx is small. We also have time steps of size Δt , and thus going from $t_0 = 0$, $t_1 = \Delta t$, and so forth. We represent the solution by the indexing scheme:

$$u(x_j, t_k) = u_j^k.$$

Like for Euler's method, we consider the approximation of the time derivative as follows:

$$\left(\frac{\partial u}{\partial t}\right)_j^k \approx \frac{u_j^{k+1} - u_j^k}{\Delta t}, \quad \text{for } 0 \leq j \leq N + 1, k \geq 0.$$

In order to approximate the second-order derivative, we will use

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j^k \approx \frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{(\Delta x)^2}, \quad 1 \leq j \leq N, k \geq 0.$$

Note that the finite difference cannot be applied directly to the first and last points ($j = 0$ and $j = N + 1$). Substituting the above finite differences into the heat equation, we see that we now have the prescription:

$$u_j^{k+1} = u_j^k + \frac{\Delta t}{(\Delta x)^2} [u_{j+1}^k - 2u_j^k + u_{j-1}^k],$$

which applies for $j = 1, 2, \dots, N$.

10.2 Boundary and initial conditions

The initial conditions are implemented by setting

$$u_j^0 = 2, \quad \text{for } j = 0, 1, 2, \dots, N + 1.$$

The boundary conditions require

$$u_0^k = 1 \quad \text{and} \quad u_{N+1}^k = 2, \quad \text{for all } k \geq 0.$$

CHAPTER 11

An example of a wine cellar problem

In the lecture (Lecture 10 in 2025), we will introduce a classic example of modelling heat flow in the ground. This will accompany the Python file `Chapter10-winecellar.ipynb` which you will be able to find on the Noteable website. We will walk you through the explicit time solver using Python, and then also the analytical solution.

CHAPTER 12

The wine cellar problem

This chapter, we will study an example of a PDE model for heat transport, and then use it to inform ourselves on a typical physical problem.

We are interested in finding the optimal depth for a storage cellar that might be used to house wine. Based on our previous discussions, if we assume that the temperature beneath the ground is given by $T(x, t)$, with $x \geq 0$ below the surface, then the temperature is governed by

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial x^2},$$

for $x \geq 0$. This is a highly averaged model, where we assume there is no transport of heat into the horizontal plane.

We shall also assume that the surface temperature is modelled by an oscillatory function. On the assumption that the temperature has been normalised so that $T = 0$ corresponds to the mean temperature of the surface, then we have

$$T(x = 0, t) = A \cos(\omega t).$$

The wavelength is then $2\pi/\omega$ which we can set to one year. So $\omega = 1/(2\pi)$ yr $^{-1}$.

In addition, we expect that the temperature should not be singular far beneath (in particular we expect it to get colder as we go further underground), at least on a scale where we are not approaching the centre of the Earth! Thus it is sensible that

$$|T(x, t)| \text{ is bounded as } x \rightarrow \infty.$$

You will not require an initial condition of the discussions below.

12.1 Numerical solutions

An example code that uses first-order explicit Euler timestepping is given in `chapter11-winecellar` on the lectures coding folder. A typical numerical output is given here:

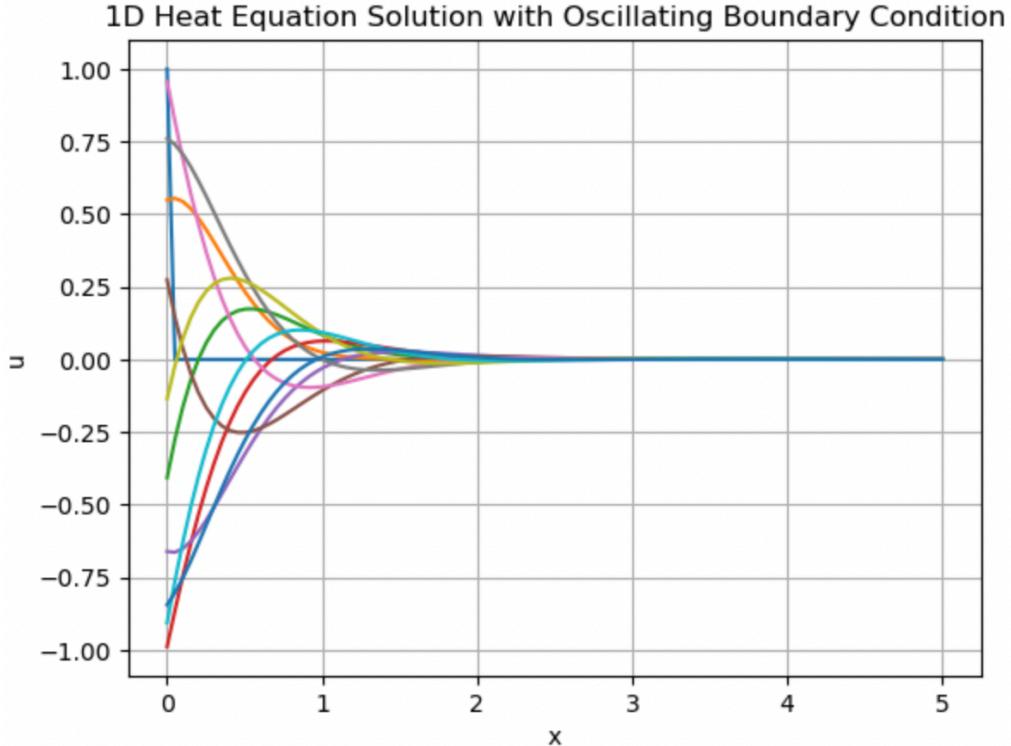


Figure 12.1: Numerical solution of the wine cellar problem

12.2 Analytical solution

Before we begin, it is useful for the purposes of manipulation to write the boundary condition as

$$T(0, t) = A\Re(e^{i\omega t}) = A\Re(\cos(\omega t) + i \sin(\omega t)) = A \cos(\omega t).$$

This has the advantage that we can manipulate exponentials much more easily.

To solve the problem analytically, we attempt a solution method called separation of variables. Let

$$T(x, t) = G(t)H(x).$$

Substitution into the PDE gives

$$G'(t)H(x) = \kappa G(t)H''(x),$$

and we attempt to separate the functions of t to one side and the functions of x to the other.

$$\frac{G'(t)}{G(t)} = \kappa \frac{H''(x)}{H(x)}$$

Now in order for two completely different functions dependent on two completely different independent variables to be equal in this way, then this implies that both sides must be equal to a constant. So we have

$$\begin{aligned} G' &= \lambda G \\ H'' &= \lambda \kappa H. \end{aligned}$$

Therefore $G = \text{const} \times e^{\lambda t}$. Because of the boundary conditions that would require $T = Ae^{i\omega t}$, you can check that $\lambda = 0$ and λ real are both impossible. The only way to

satisfy the boundary conditions is if $\lambda = i\omega$. So we conclude that

$$G(t) = \text{const.} \times e^{i\omega t}.$$

To solve

$$H'' = (i\omega/\kappa)H,$$

you can attempt the substitution $H(x) = e^{rx}$ and solve the resultant equation for the characteristic polynomial in r . You will do this in the problem set, and obtain

$$r = \pm\sqrt{i}\sqrt{\omega/\kappa} = \pm\sqrt{\omega/\kappa}\frac{1+i}{\sqrt{2}}.$$

Therefore

$$H(x) = C_1 e^{\sqrt{\omega/(2\kappa)}x} e^{\sqrt{\omega/(2\kappa)}ix} + C_2 e^{-\sqrt{\omega/(2\kappa)}x} e^{-\sqrt{\omega/(2\kappa)}ix}.$$

The first solution should be discarded since it blows up as $x \rightarrow \infty$.

Finally, we conclude that the solution is given by

$$T = e^{i\omega t} e^{-\sqrt{\omega/(2\kappa)}x} e^{-\sqrt{\omega/(2\kappa)}ix}.$$

Move the imaginary arguments together:

$$T = \text{const.} \times e^{-\sqrt{\omega/(2\kappa)}x} e^{i(-\sqrt{\omega/(2\kappa)}x + \omega t)}.$$

Now we take the real part, giving

$$T = A e^{-\sqrt{\omega/(2\kappa)}x} \cos(-\sqrt{\omega/(2\kappa)}x + \omega t), \quad (12.1)$$

and we see that having set the constant to A the solution indeed satisfies $T(0, t) = A \cos(\omega t)$.

12.3 Selection of optimal depth

The optimal depth of the wine cellar is selected by taking the depth to be such that the cosine is completely out of phase with the surface solution. This means that

$$-\sqrt{\omega/(2\kappa)}x + \omega t + \pi = \omega t.$$

You can substitute other multiples of π . Thus we have

$$x = \pi \sqrt{\frac{2\kappa}{\omega}}.$$

We can estimate the numbers as follows. Converting from 1 year to seconds yields

$$\omega = \frac{2\pi}{3.15 \times 10^7 \text{ s}}.$$

Also, the κ for dry soil is around $0.002 \text{ cm}^2/\text{s}$.

Putting these numbers in yields

$$x \approx 445 \text{ cm} = 4.45 \text{ m}.$$

Part III

Energy balance models

CHAPTER 13

EBM with nonlinear albedo

Recall that we previously introduced the basic energy balance model in Chapter 4. There, we derived the basic heat equation model (Equation 4.7) for the Earth's temperature given by the following ordinary differential equation (ODE) for $T = T(t)$,

$$C \frac{dT}{dt} = Q(1 - a) - \sigma\gamma T^4, \quad (13.1)$$

where we have defined $C = \rho c_p d$ as the heat capacity of the atmosphere. Above, the solar flux, Q , is often taken to be $Q = 1370/4 = 342 \text{ W/m}^2$, $\sigma = 5.67 \times 10^{-8} \text{ W}/(\text{m}^2\text{K}^4)$, and $\gamma \leq 1$ is the greenhouse gas factor.

In this chapter, we discuss the effects of considering a nonlinear albedo, $a = a(T)$, as well as some of the numerical and analytical tools at our disposal for studying the above equation.

13.1 Steady-state analysis

Below, we shall let $T = T_\infty$ be the steady-state solution that is independent of time. Whenever convenient, we will drop the subscript notation and simply refer to the steady state as T .

Previously, we have assumed that the planetary albedo, a , is constant and independent of temperature. In actuality, water can turn to snow and ice and vice versa; since snow and ice have much higher albedo than open water, then we should consider $a = a(T)$.

Let us assume that there are two relevant ranges to consider: $T < 150\text{K}$ (cold) and $T > 280\text{K}$ (hot). Let us assume that the albedo is, in these two regions:

$$a(T) \approx \begin{cases} 0.7 & \text{if } T < 150\text{K}, \\ 0.3 & \text{if } T > 280\text{K}. \end{cases}$$

The above guarantees that more energy is reflected if temperatures are low. To model this process, we can use a ramp function to specify the albedo over all temperatures:

$$a(T) = A - B \tanh(k(T - 265)). \quad (13.2)$$

where $A = 0.5$, $B = 0.2$, $k = 0.1$, and $T_0 = 265\text{K}$. Recall that the tanh function is given by

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}.$$

Let us further assume that the system is in steady state, so that the temperature is determined by solving the equation

$$f(T) = Q[1 - a(T)] - \sigma\gamma T^4 = 0. \quad (13.3)$$

In the following code, we plot the two terms that make up f , and their intersections indicate roots of $f = 0$. We then use the Python ‘fsolve’ function to approximate the roots given initial guesses.

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize as sciopt

Q = 342
sigma = 5.67e-8
gamma = 0.62

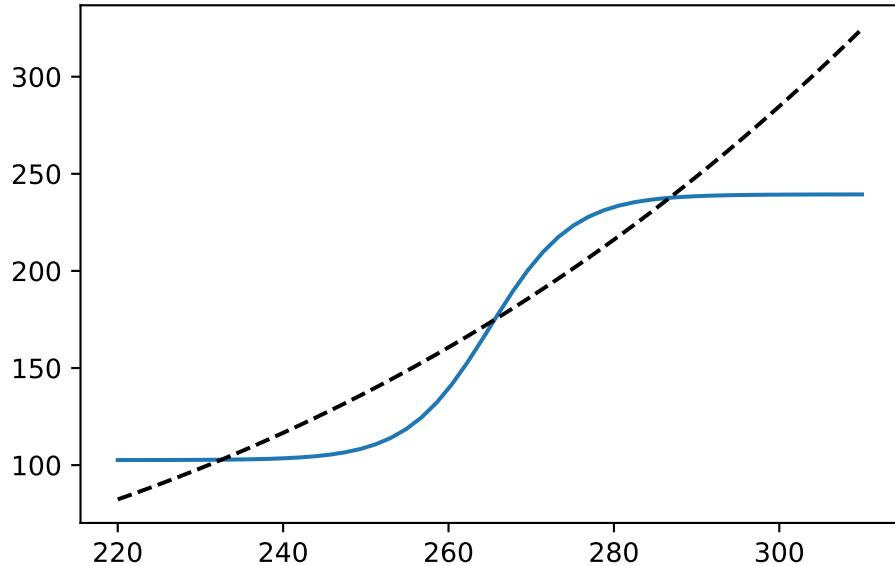
TT = np.linspace(220,310,50)

def fun(T):
    a = 0.5 - 0.2*np.tanh((T - 265)/10)
    x = (1-a)*Q
    return x
LHS = fun(TT)

plt.plot(TT, LHS)
plt.plot(TT, gamma*sigma*TT**4, 'k--')

def eq(T):
    x = fun(T) - gamma*sigma*T**4
    return x
T1 = sciopt.fsolve(eq, 230)
T2 = sciopt.fsolve(eq, 265)
T3 = sciopt.fsolve(eq, 290)
print("T1 = {:.2f}".format(T1[0]))
print("T2 = {:.2f}".format(T2[0]))
print("T3 = {:.2f}".format(T3[0]))
```

T1 = 232.55
T2 = 265.56
T3 = 286.74



Therefore multiple equilibria are observed.

13.2 Bifurcation diagram

During the lectures, the above script plotting the steady-state solutions will be examined in order to understand the effects of changing Q .

It will be observed that dependent on the solar radiation, there may be one or three steady-state solutions. This leads to the following key bifurcation diagram.

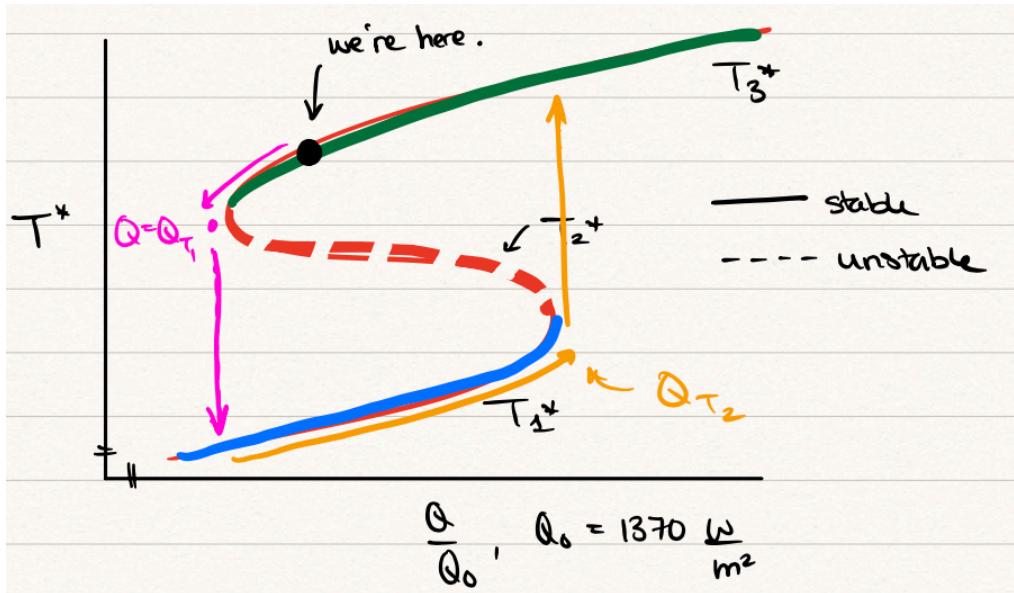


Figure 13.1: Bifurcation diagram of Q/Q_0 vs T_∞

We noted the following:

- The system has three steady states, given by the green, red, and blue curves.
- The middle state is unstable (shown dashed).
- The system exhibits hysteresis. Note that if we decrease Q/Q_0 past the tipping point, marked Q_{T1} in the image, then we would evolve to the lower stable steady

state (which is the ice state). However, while we are in the ice state, if we were to attempt to increase the solar radiation to return to the green branch, we would need to arrive at Q_{T_2} to do so; this irreversibility is known as hysteresis. ## Dynamics and phase line solutions

The full time-dependent model is given by

$$C \frac{dT}{dt} = f(T),$$

so we may use the positivity or negativity of f in order to sketch the time-dependent behaviour of the system.

To see this, we can perform an asymptotic analysis near the fixed points. Let the initial condition be considered near the fixed point:

$$T(t=0) = T_\infty + \delta,$$

where $\delta \ll 1$. Then we expand the solution into an asymptotic expansion,

$$T(t) = T_\infty + \delta T_1(t) + \delta T^2(t) + \dots$$

Substitution into the above ODE gives, at $O(\delta)$,

$$C \frac{dT_1}{dt} = f'(T_\infty) T_1,$$

and hence, with $T_1(0) = 1$, we have

$$T_1(t) = e^{f'(T_\infty)t/C}.$$

Therefore, depending on the positivity or negativity of the gradient function, the perturbation will either decay or grow as $t \rightarrow \infty$.

It can then be verified that the centre equilibria is unstable while the other two are stable. The higher temperature corresponds to the one that the Earth is currently in, but according to this model, there seems to be the possibility of a colder climate (50 degrees colder) where the Earth is entirely covered with snow and ice.

Interestingly, there is some evidence that the Earth's climate may have been in this so-called *Snowball Earth* state up to four times between 750 million and 580 million years ago (Neoproterozoic age). Observations of geological deposits suggest that the Earth has undergone periods of complete global glaciation where there have been very minimal biological activity. During this period, there is a massive build-up of CO₂ in the atmosphere, leading to huge greenhouse effect. As γ decreases in our model, the equilibrium can then shift, suddenly transitioning the Planet into the warm state.

13.3 Re-scaling and Budyko's model

To model the outgoing radiation, we use the quartic Stefan-Boltzmann law. However, over the range of temperatures we are interested-in, it seems that a simpler approximation is sufficient. In your homework, you will investigate the re-scaling and shifting of temperature, such that

$$T = T_0 + [T]\tilde{T},$$

where $T_0 = 265\text{K}$. Then under the assumption that temperatures are not-so-far from T_0 , we expand

$$(T_0 + [T]\tilde{T})^4 \sim T_0^4 + 4T_0^3[T]\tilde{T} = C_1 + C_2\tilde{T}.$$

This simplifies the model considerably.

This is, in fact, a theoretical justification of **Budyko's model**, first suggested by Budyko ([Budyko, 1969](#)), where instead of the Stefan-Boltmann law that provides the cumbersome quartic power of T , we use instead

$$E_{\text{out}} = \sigma\gamma T^4 \approx A + BT,$$

where A and B will vary with location and climate. For instance, for the Northern Hemisphere, ([Kaper und Engler, 2013](#)) gives the values of $A = 203.3 \text{ Wm}^{-2}$ and $B = 2.09 \text{ Wm}^{-2} \text{ deg}^{-1}$ and where temperature is measured in degrees Celcius. As shown above, one can interpret this as a linear expansion of the Stefan-Boltzmann law (after which similar values of A and B are derived).

The above Budyko model will be a staple of the latitude-dependent model introduced in the next chapter.

CHAPTER 14

EBM with latitude I

We now introduce a model that allows for two additional effects: an albedo that models ice dynamics and latitude variation; and a transport term, which allows for heat movement (primarily convection).

To begin, let us use a spherical coordinate system where φ is the latitude angle, with $\varphi = 0$ at the equator and $\varphi = \pi/2$ at the North Pole. Note that this differs from the traditional spherical coordinate system where ϕ is the polar angle. It is moreover convenient to use

$$y = \sin \varphi, \quad y \in [-1, 1], \quad \varphi \in [-\pi/2, \pi/2].$$

Our conservation equation takes the general form for the temperature, $T(y, t)$:

$$C \frac{\partial T}{\partial t} = E_{\text{in}} - E_{\text{out}} + E_{\text{transport}}, \quad t > 0 \quad (14.1)$$

where we shall use the following:

$$\begin{aligned} E_{\text{in}}(y, t) &= Qs(y)(1 - \alpha(y)), \\ E_{\text{out}}(y, t) &= A + BT, \\ E_{\text{transport}}(y, t) &= k(\bar{T} - T), \end{aligned} \quad (14.2)$$

Each of these terms we explain below in turn.

The above equation will be supplemented with initial conditions:

$$T(y, 0) = T_i(y).$$

14.1 Incoming energy

We make the following remarks about E_{in} .

- Notice that now, compared to the previous model, we have weighted the incoming solar constant Q with a term $s(y)$ and also our albedo no longer directly depends on temperature but instead depends on the latitude.
- Points on Earth that are closer to the equator receive more direct sunlight and experience more hours of daylight on average. We thus define a function $s(x)$ to account for this and consider the incoming radiation as weighted by $Qs(y)$.

- Note that the total solar input is calculated by a surface integral:

$$\int_{\theta=0}^{\theta=2\pi} \int_{\varphi=-\pi/2}^{\pi/2} Q s(y = \sin \varphi) (a^2 \cos \varphi) d\varphi d\theta = 4\pi a^2 Q \int_0^1 s(y) dy = 4\pi a^2 Q,$$

if the function $s(y)$ has been normalised appropriately so that its integral is equal to one, and furthermore the radiation is assumed symmetric about the equator.

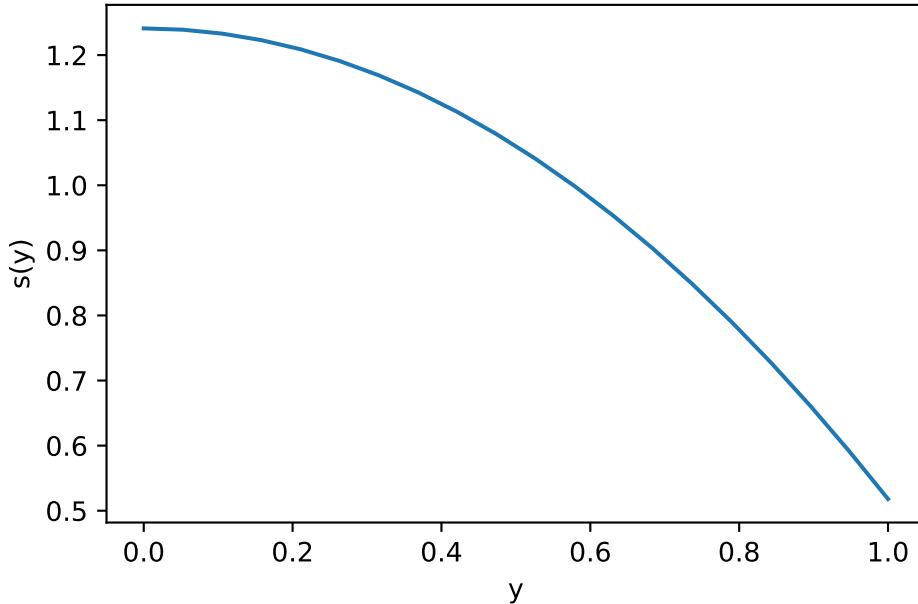
- The above is equal to the solar flux intercepted by an area of the circular disk of the earth seen by the sun, which is $\pi a^2 S$ for $S = 1370 \text{ W/m}^2$. Therefore $Q = S/4 \approx 342 \text{ W/m}^2$.

Finally the function $s(y)$ is fitted to data. We can assume it to be given by

$$s(y) = 1 - S_2 P_2(y), \quad S_2 = 0.482, \quad P_2(y) = (3y^2 - 1)/2. \quad (14.3)$$

```
import numpy as np
import matplotlib.pyplot as plt

y = np.linspace(0, 1, 20)
plt.plot(y, 1 - 0.482*(3*y**2 - 1)/2)
plt.xlabel('y'); plt.ylabel('s(y)');
```



We assume that an ice sheet forms if the temperature is sufficiently low, and $T < T_c = 10^\circ\text{C}$. The ice forms at an ice line, where $y = y_s$. We then have the following form from the albedo:

$$a(y) = \begin{cases} a_i = 0.62 & y > y_s, \\ a_w = 0.32 & y < y_s, \\ \frac{1}{2}(a_i + a_w) & y = y_s. \end{cases}$$

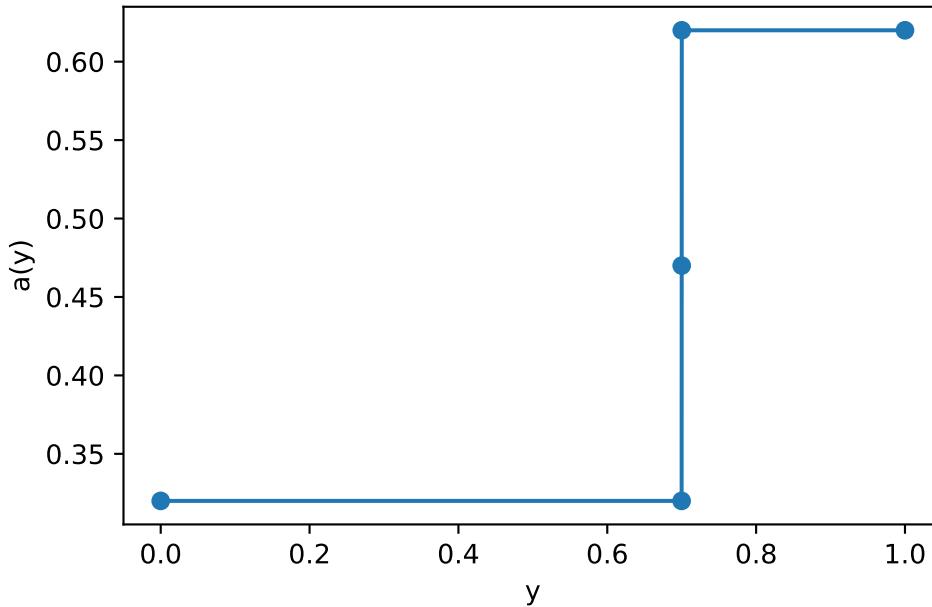
Therefore we are using a discontinuous model for the albedo. Note that the ice boundary y_s is an unknown to be determined by the model.

```

import numpy as np
import matplotlib.pyplot as plt

ys = 0.7
y = np.array([0, ys-0.0001, ys, ys+0.0001, 1])
ai = 0.62; aw = 0.32;
a = np.piecewise(y, [y < ys, y == ys, y > ys], [aw, 0.5*(aw+ai), ai])
plt.plot(y, a, '-o')
plt.xlabel('y'); plt.ylabel('a(y)');

```



The above forms what is referred to as a free-boundary problem, as the boundary $y = y_s$ is solved as part of the model. Because y_s is essentially determined by where the temperature reaches the critical value, i.e. $T(y_s, t) = T_c$, then the above albedo not only depends on latitude, but also on the current temperature distribution as well.

14.2 Outgoing energy

The outgoing energy is approximated via the linear Budyko approximation discussed in Section 13.3.

$$E_{\text{out}} = A + BT,$$

where $A = 202 \text{W/m}^2$ and $B = 1.9 \text{W}/(\text{m}^2\text{C})$. These figures are taken from a 1993 paper by [Graves, Lee, and North](#).

14.3 Transport energy

The next term we should consider is the transport. In the case of this simple model, transport is taken to be modelled via a “Newton law of cooling” type expression. We assume that

$$\text{rate of change of heat energy} = k(\bar{T} - T),$$

where it can be verified (exercise) that the global mean temperature is the same as the hemispherically averaged temperature, i.e.

$$\bar{T} = \frac{1}{\text{surface area}} \iint T S = \int_0^1 T(y) dy.$$

Above, we shall use a value of $k = 1.6B$, again estimated based on data and studies.

This completes our review of all the components (Equation 14.2) that make up the EBM model (Equation 14.1).

14.4 Steady-state temperature

Let $T_\infty(y)$ be the steady-state temperature and \bar{T}_∞ be the corresponding global mean temperature. Then from setting the LHS of (Equation 14.1) to zero, we have the fact that T_∞ is given by solving the implicit equation:

$$T_\infty(y) = \Phi(T_\infty) = \frac{k\bar{T}_\infty + Qs(y)(1 - \alpha(y)) - A}{B + C}.$$

It is important to remember that the albedo, $\alpha(y)$, implicitly depends on T_∞ because at each location y , we must decide if the temperature is less or greater than the threshold T_c , which is the temperature of the ice line.

The above equation is somewhat tricky because of its implicit nature; it is made even trickier by virtue that multiple equilibria exist at the same parameter values. We will begin studying this problem, but before doing so, let us make a few remarks:

- It is possible to solve for the steady-state global mean, \bar{T}_∞ , as a function of the ice line location, y_s .
- It is possible to solve for the ice line location, y_s , although not entirely explicitly. Up to four solutions are possible.
- Once the ice line is known it is possible to solve for the steady-state.

The next few chapters will investigate the above steady-state properties. However note that a numerical solution can be developed without the need to go through the above procedures, as well, albeit in a complicated fashion (In 2024, this was addressed during the Lecture 25; see the visualiser notes).

CHAPTER 15

EBM with latitude II

This lecture, we continue our investigation of the latitude-dependent EBM.

Latitude-dependent EBM

Our conservation equation takes the general form for the temperature, $T(y, t)$:

$$C \frac{\partial T}{\partial t} = E_{\text{in}} - E_{\text{out}} + E_{\text{transport}}, \quad t > 0 \quad (15.1)$$

where we shall use the following:

$$\begin{aligned} E_{\text{in}}(y, t) &= Qs(y)(1 - a(y)), \\ E_{\text{out}}(y, t) &= A + BT, \\ E_{\text{transport}}(y, t) &= k(\bar{T} - T), \end{aligned} \quad (15.2)$$

Our goal is to study the steady-state solutions, $T = T_\infty(y)$, given by solving the implicit equation:

$$T_\infty(y) = \Phi(T_\infty) = \frac{k\bar{T}_\infty + Qs(y)(1 - a(y)) - A}{B + k}. \quad (15.3)$$

We want to know whether other solutions exist with ice lines at $y_s \in [0, 1]$.

15.1 Development of an equation for the ice line

It turns out to be possible to develop an equation for the ice line. First, note from your problem set in Chapter 29, you find that it is possible to solve for the steady-state mean temperature, which we write as \bar{T}_∞ . In your problem set, you will develop the following equation by integrating over (Equation 15.1) from $y = 0$ to $y = 1$.

Taking the steady-state limit of (Equation 36.1) this yields

$$\bar{T}_\infty = \frac{Q(1 - \bar{a}) - A}{B}. \quad (15.4)$$

The quantity \bar{a} is given by integrating the albedo as follows:

$$\bar{a} = \int_0^1 s(y)a(y) dy = a_i + (a_w - a_i)y_s[1 - 0.241(y_s^2 - 1)], \quad (15.5)$$

which is given by (Equation 36.2). You may check that the above formula has the right signs by considering either the complete ice case (ice line at the equator $y_s = 0$) or the complete water/land case (ice line at $y_s = 1$).

Now we return to the implicit equation for the temperature, which is given by (Equation 15.3) and repeated here:

$$T_\infty(y) = \Phi(T_\infty) = \frac{k\bar{T}_\infty + Qs(y)(1 - a(y)) - A}{B + k}.$$

Substitute the mean temperature in, and this now yields

$$T_\infty(y) = \frac{Q}{B + k} \left[s(y)[1 - a(y)] + \frac{k}{B}(1 - \bar{a}) \right] - \frac{A}{B}. \quad (15.6)$$

This ice line is then found by setting $T = T_C$ at $y = y_s$ in the above formula. Notice, though, that since $s(y)$ is a cubic function, then we would need to solve a cubic equation in general.

15.2 A word about the parameter space

Solutions to our latitude-dependent EBM can be symbolically written as follows:

$$T = T(t, y) = G(y; A, b, k, C, Q, a_i, a_w).$$

As you can see, even though it is a relatively simple equation in the sense it is only an ODE in time (and does not involve any spatial derivatives in y), is still complicated because the behaviour of the system can depend in a non-trivial way on all the parameters.

You can schematically think of the solution space as being plotted in 8-dimensional space (or even higher, since the albedo a can be specified more generally). So for instance, bifurcation diagrams can then be plotted for some norm of the solution, say \bar{T} versus the seven other parameters.

There are other representations of the bifurcation diagram(s). For example, you might plot y_s vs Q or \bar{T} vs A , and so forth.

15.3 A bifurcation diagram for Q vs y_s

Returning to the ice line, we are interested in keeping all other parameters fixed, and then attempting to understand how the ice line evolves as the solar constant Q is changed. For example, we might believe that as Q increases (and hence temperatures rise), the ice line will move towards the North Pole. And as Q decreases, the ice line moves towards the equator.

We return to (Equation 15.6) and consider inverting the formulation. For each given ice line location, $y_s \in (0, 1)$, we solve for the Q value. This gives

$$Q = K(y_s) \equiv \frac{\left(T_C + \frac{A}{B} \right) (B + k)}{s(y_s)[1 - \frac{1}{2}(a_i + a_w)] + \frac{k}{B}(1 - \bar{a})}. \quad (15.7)$$

Above, we have used the fact that $a(y_s) = \frac{1}{2}(a_i + a_w)$. This can now be plotted in Python, either in the notebook written during lectures, or via the code below.

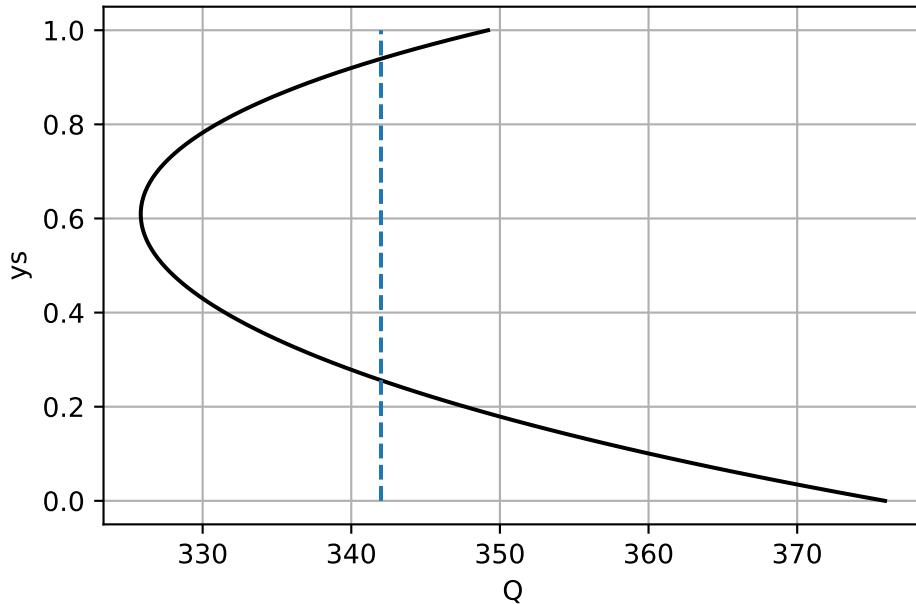
```

import numpy as np
import matplotlib.pyplot as plt
A = 202 # outgoing radiation
B = 1.9 # outgoing radiation
k = 1.6*B # transport parameter
s = lambda y: 1 - 0.482*(3*y**2 - 1)/2 # solar weighting
aw = 0.32 # water albedo
ai = 0.62 # ice albedo
Tc = -10.0 # critical temperature for ice formation
Q0 = 342.0 # solar constant (1370 W/m^2 divided by 4)

abar = lambda ys: ai + (aw - ai)*ys*(1 - 0.241*(ys**2 - 1))
Qfunc = lambda ys: (Tc + A/B)*(B+k)/(s(ys)*(1 - (ai+aw)/2) + k/B*(1 - abar(ys)))

# Solve for the ice line
ys = np.linspace(0, 1, 100);
Qs = Qfunc(ys);
plt.plot(Qs, ys, 'k')
plt.plot([Q0, Q0], [0, 1], '--')
plt.xlabel('Q');
plt.ylabel('ys');
plt.grid(1)

```



On the above graph, we have plotted the reference line, which is at $Q = 342 \text{ W-m-sq}$. So amazingly, there are two intersections with the black curve, which in fact indicates that two ice lines seem to be possible. One ice line as y_s near 0.95 and the other has y_s much lower down and near the equator.

Moreover, the above graph is not complete! When we computed it, we specified that $a(y_s) = \frac{1}{2}(a_i + a_w)$ but this would not be true if there were no ice line, or equivalently if the ice line is located directly at $y = 1$ or $y = 0$. These yield the so-called ice-free states and ice-covered states, respectively.

15.4 Ice-free state

The ice-free state is the state for which $a = a_w = \bar{a}$ for all $y \in [0, 1]$. In this case, the solution can be directly calculated from Equation 15.6. It would then be given by

$$T_\infty(y) = \frac{Q(1 - a_w)}{B + k} \left[s(y) + \frac{k}{B} \right] - \frac{A}{B}.$$

This solution has a requirement, which is that when considering the solar constant Q , the solar constant cannot be so weak so that an ice line appears for $y < 1$. The minimum value of Q is therefore determined by pinning the ice line right at the North Pole. Thus

$$Q > \frac{(B + k)(T_C + A/B)}{(1 - a_w)[s(1) + k/B]} \approx 330 \text{ W m}^{-2}.$$

It is of interest to verify that the global mean temperature for a system in this state is a warm 16 degrees Celsius. According to this model, then, if the mean temperature is above this value, it is possible to have a state where there is no ice anywhere!

15.5 Ice-covered state

A similar argument applies setting $a = a_i = \bar{a}$ for all $y \in (0, 1]$. In this case,

$$T_\infty(y) = \frac{Q(1 - a_i)}{B + k} \left[s(y) + \frac{k}{B} \right] - \frac{A}{B}.$$

This time, there is a maximal condition on Q such that any higher value would require an ice line somewhere within the domain, i.e. $y > 0$. Thus

$$Q < \frac{(B + k)(T_C + A/B)}{(1 - a_i)[s(0) + k/B]} \approx 441 \text{ W m}^{-2}.$$

Again we may verify that this corresponds to a mean temperature of -38 degrees Celsius.

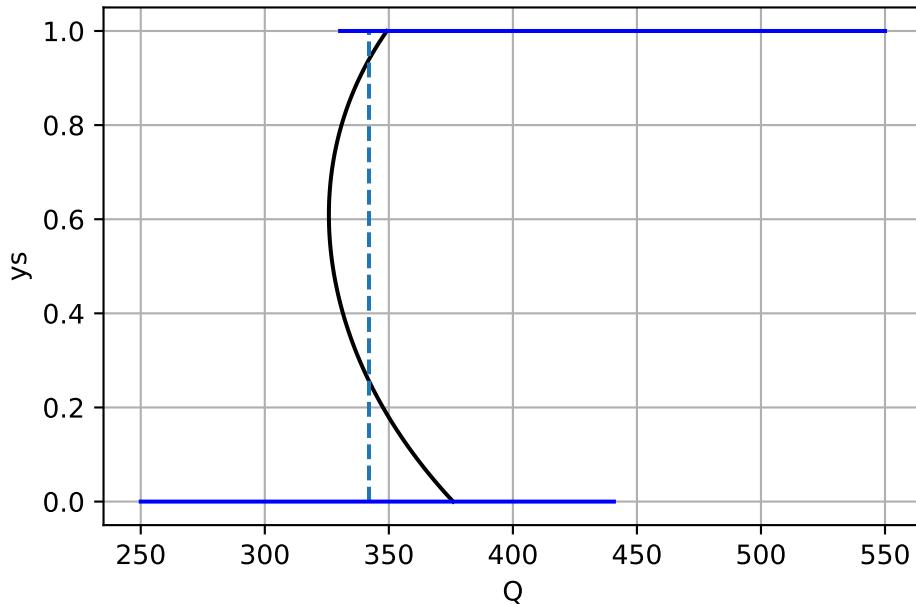
```
import numpy as np
import matplotlib.pyplot as plt
A = 202 # outgoing radiation
B = 1.9 # outgoing radiation
k = 1.6*B # transport parameter
s = lambda y: 1 - 0.482*(3*y**2 - 1)/2 # solar weighting
aw = 0.32 # water albedo
ai = 0.62 # ice albedo
Tc = -10.0 # critical temperature for ice formation
Q0 = 342.0 # solar constant (1370 W/m^2 divided by 4)

Qmin = ((B+k)*(Tc + A/B))/((1-aw)*(s(1)+k/B))
Qmax = ((B+k)*(Tc + A/B))/((1-ai)*(s(0)+k/B))
print("Minimal Q for ice-free state = ", Qmin)
print("Max Q for ice-covered state = ", Qmax)

# Note version in lectures was a 1/2 factor off on the second grouping of terms;
# now corrected to match Q6 of PS4
abar = lambda ys: ai + (aw - ai)*ys*(1 - 0.241*(ys**2 - 1))
Qfunc = lambda ys: (Tc + A/B)*(B+k)/(s(ys)*(1 - (ai+aw)/2) + k/B*(1 - abar(ys)))
```

```
# Solve for the ice line
ys = np.linspace(0, 1, 100);
Qs = Qfunc(ys);
plt.plot(Qs, ys, 'k')
plt.plot([Q0, Q0], [0, 1], '--')
plt.plot([250, Qmax], [0, 0], 'b')
plt.plot([Qmin, 550], [1, 1], 'b')
plt.xlabel('Q');
plt.ylabel('ys');
plt.grid(1)
```

Minimal Q for ice-free state = 330.3616063989335
 Max Q for ice-covered state = 440.72694936919913



15.6 Partially ice-covered states

In lectures, we demonstrated, that there are four possible solutions of the latitude EBM. Two of the solutions, developed above, correspond to completely covered ice-state and ice-free states. The other two solutions are partially-iced states.

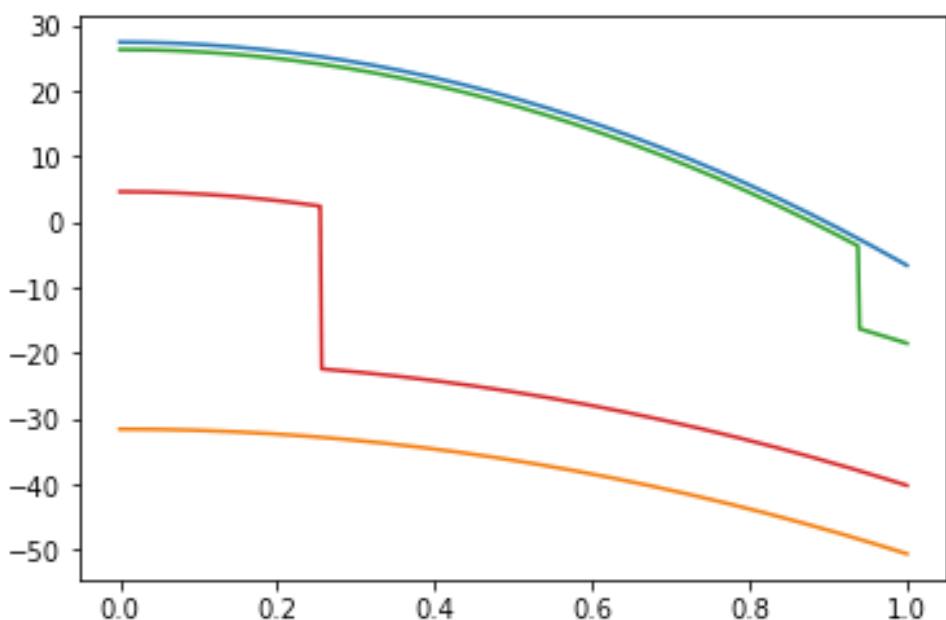


Figure 15.1: The four possible solutions; orange and blue show the completely-ice and ice-free states; red and green show the partial ice-covered state. These are shown for $A = 202$, $B = 1.9$, $k = 1.6B$, and $Q = 342\$$

CHAPTER 16

EBM with latitude III

We continue our investigation of the latitude-dependent model of Chapter 15.

16.1 Studying the mean temperature

Following the previous chapter, we now have a better understanding of the relationship between the ice line, y_s , and the solar forcing, Q . We demonstrated that for the baseline case of $Q = 342$, (at least) four solutions are possible.

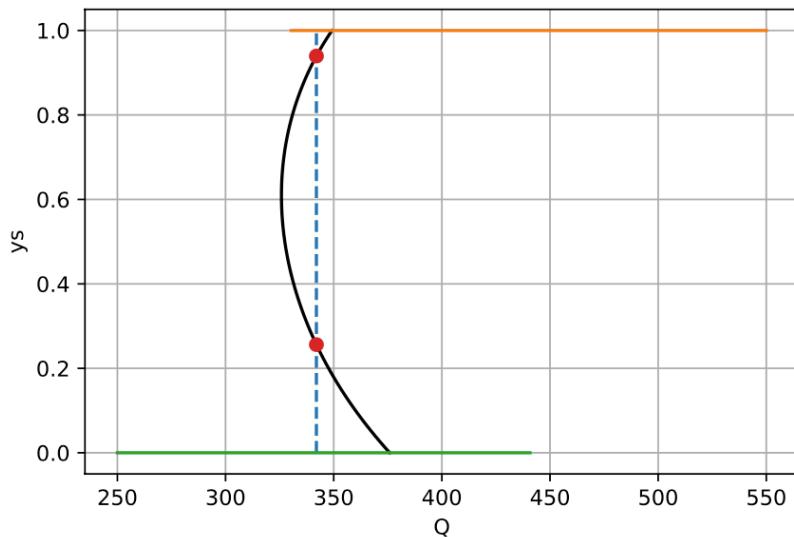


Figure 16.1: The iceline

Examination of the root shows that the two non-trivial ice lines are approximately at $y_s = 0.256$ and $y_s = 0.939$ for $Q = 342$. Using the script developed in lectures, we can study a similar graph of the mean temperature versus the solar forcing.

This yields the following picture:

During the lecture, we discussed the so-called *Snowball Earth* scenario where decreasing the solar constant may cause the stable state of the planet to jump down to the green branch shown above (completely frozen Earth). The solution exhibits *hysteresis* which describes the non-reversibility shown in the bifurcation diagram. This will be further expounded in the following discussion on stability.

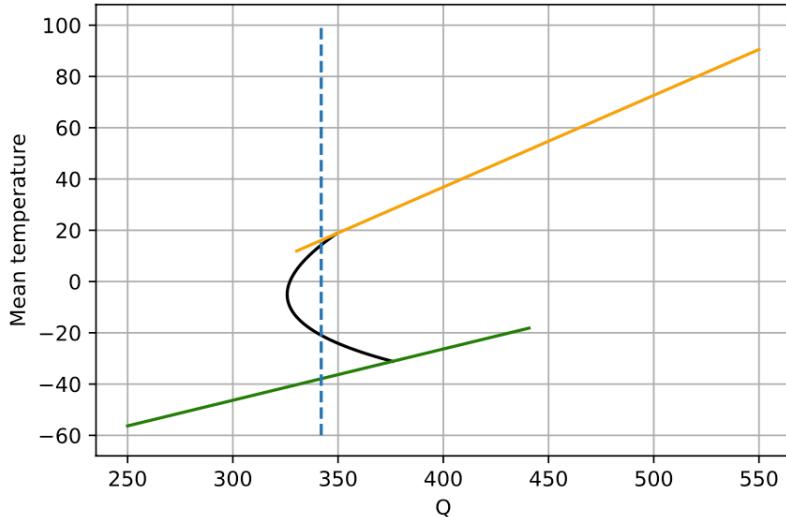


Figure 16.2: Mean temperature

16.2 Stability

So far, we have only discussed the steady-state solutions of the latitude-dependent EBM, repeated below:

$$C \frac{\partial T}{\partial t} = Qs(y)[1 - a(y)] - (A + BT) + k(\bar{T} - T).$$

In this section, we investigate the stability analysis for the mean temperature, \bar{T} . This was given in Equation 36.1 and repeated below:

$$C \frac{d\bar{T}}{dt} = G(\bar{T}) \equiv Q(1 - \bar{a}) - (A + B\bar{T}), \quad (16.1)$$

The following linearisation argument was presented by [Cahalan and North \(1979\)](#), with some additional details to avoid confusion. Let us linearise this about a steady-state solution, writing

$$\bar{T} = \bar{T}^* + u(t),$$

where $u \ll 1$. By Taylor series,

$$G(\bar{T}) = G(\bar{T}^*) + G'(\bar{T}^*)u + O(u^2),$$

so substitution into the ODE yields the linear equation

$$C \frac{du}{dt} = G'(\bar{T}^*)u. \quad (16.2)$$

We need to calculate the derivative term on the RHS. First, using the equation for G in Equation 16.1, we have

$$G'(\bar{T}^*) = -B - Q \frac{d\bar{a}}{d\bar{T}} \Big|_{\bar{T}=\bar{T}^*}. \quad (16.3)$$

The above calculation assumes that Q is a fixed number, as it should, since Q is regarded as a parameter (we choose a value of Q , then this allows us to evolve the ODE).

Calahan & North's "trick"

The following presentation took about 3 lines of text in the presentation of Cahalan & North (1979), between eqns (1.2) and (1.3) in the image below. Students have inquired about the manipulations.

In order to examine the linear stability, let $T_0 = T_0^*$ + $\delta(t)$, where T_0^* is a solution to the steady-state problem. Then to the first order in $\delta(t)$,

$$C\dot{\delta}(t) + (I' - Q\bar{a}')\delta(t) = 0, \quad (1.2)$$

where a prime denotes the derivative, evaluated at $T_0 = T_0^*$. The stability is determined by the sign of the expression in parentheses. This sign can be expressed in terms of the slope of the solution curve $Q(T_0)$, as follows. Differentiating the steady-state equation $I = Q\bar{a}$ and substituting, Eq.(1.2) becomes

$$C\dot{\delta}(t) + \frac{dQ}{dT_0} \bar{a}\delta(t) = 0. \quad (1.3)$$

This last equation embodies the "slope-stability" theorem

$$\left. \begin{array}{l} \frac{dQ}{dT_0} > 0 \leftrightarrow \text{stability} \\ \frac{dQ}{dT_0} < 0 \leftrightarrow \text{instability} \end{array} \right\}. \quad (1.4)$$

Figure 16.3: From Cahalan & North, p.1179

The problem is what to do with the derivative of \bar{a} that appears in (Equation 16.3). This is a strange quantity, since \bar{a} does not explicitly depend on \bar{T} , and is given in (Equation 15.5). The issue is that there is a hidden dependence: \bar{a} depends on the ice line, y_s , which depends on not only Q , but is multivalued as well!

Cahalan & North, whether knowingly or unknowingly, applied a trick, which is to consider that Q is not a parameter, but that it can be viewed as depending on \bar{T}^* , the steady state (this dependence is shown in Figure 16.2). Therefore, Cahalan & North returned to the equation (Equation 15.4) that gives the steady-state:

$$T^* = \frac{Q(1 - \bar{a}) - A}{B}.$$

We differentiate this equation with respect to \bar{T}^* , but now assume both Q and \bar{a} depend on \bar{T}^* . This yields

$$1 = \frac{1}{B} \frac{dQ}{d\bar{T}^*} (1 - \bar{a}) - \frac{Q}{B} \frac{d\bar{a}}{d\bar{T}^*} \implies B = \frac{dQ}{d\bar{T}^*} (1 - \bar{a}) - Q \frac{d\bar{a}}{d\bar{T}^*} \quad (16.4)$$

We substitute the above value of B into (Equation 16.3), and then finally into the equation for the linearisation (Equation 16.2). Then we have

$$C \frac{du}{dt} = - \left[(1 - \bar{a}) \frac{dQ}{d\bar{T}^*} \right] u(t) = -\gamma u(t),$$

where the constant γ has been defined by the above. Thus, the solution is given by

$$u(t) = u(0) e^{-\gamma t/C}.$$

Note then that the crucial quantity is

$$\gamma \equiv (1 - \bar{a}) \frac{dQ}{d\bar{T}^*}.$$

If γ is positive, the equilibrium point is stable, while if γ is negative, the equilibrium is unstable. Because $(1 - \bar{a}) > 0$, then this yields the so-called slope-stability theorem coined by [Cahalan and North \(1979\)](#):

$$\frac{dQ}{d\bar{T}^*} > 0 \implies \text{stable},$$

$$\frac{dQ}{d\bar{T}^*} < 0 \implies \text{unstable}.$$

We may now plot the bifurcation diagram, shown in (Q, \bar{T}) -space. First, return to Figure 16.2. In lecture 21, we drew the following picture:

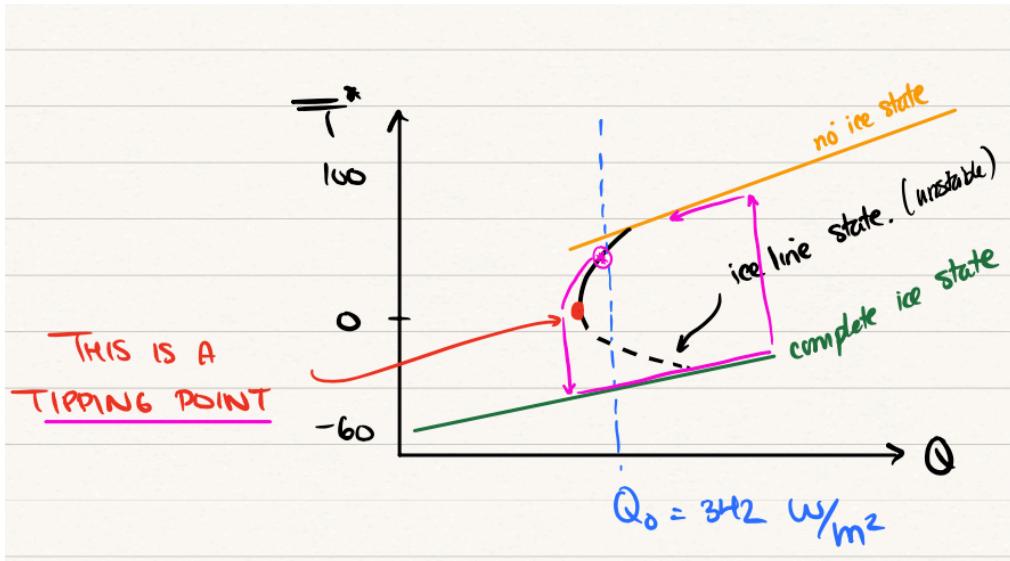


Figure 16.4: Bifurcation diagram for (Q, \bar{T})

Note the following characteristics.

1. There are up to four steady states (green, orange, two blacks).
2. Those branches with positive gradient are stable; there is only one branch with a negative gradient and it is thus unstable. In the image, these are shown solid for stable and dashed for unstable.
3. The diagram indicates tipping points and hysteresis. As explained in the lectures, one can encounter a situation where, beginning from the upper orange branches, the solar radiation is decreased past the tipping point. The solution must then evolve to the lowermost stable state (green). However, increasing Q does not return us to the original non-frozen state—unless Q is increased by a massive amount. This ‘hysteresis’ loop is shown with the maroon colours in the image above.

CHAPTER 17

EBM with latitude IV

Note

This material has been added and will be covered in problem class on 27 Mar 2025.

It is possible to solve the latitude-dependent EBM (steady-state) in a slightly more direct way (one that does not require making algebraic simplifications). We can do this by using a nonlinear solver such as Newton's method. In essence, we are seeking the solution to:

$$T = \Phi(T) = \frac{Qs(y)[1 - a(T)] - A + k\bar{T}}{B + k}$$

where in the above, $T(y)$ replaces $T_\infty(y)$.

In your problem class, you will go through the design of the following Python code, which is also found in the code repository.

First, the appropriate constants are defined.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import fsolve

A = 202 # outgoing radiation
B = 1.9 # outgoing radiation
k = 1.6*B # transport parameter
s = lambda y: 1 - 0.482*(3*y**2 - 1)/2 # solar weighting
aw = 0.32 # water albedo
ai = 0.62 # ice albedo
Tc = -10.0 # critical temperature for ice formation
Q = 342.0 # solar constant (1380 W/m^2 divided by 4)
```

Here we define the albedo function $a(y)$ and also the function that we would seek to solve by Newton's method

```
def afunc(y, ys):
    # Non-smooth albedo function
    # We want to make 'a' same dimensions as 'y'. Occasionally there are issues if we
```

```

y = np.array(y, ndmin=1) # Converts scalars to arrays but keeps arrays unchanged
a = np.zeros_like(y) # Same shape as y

a = 0*y
for i, yy in enumerate(y):
    if yy > ys:
        aa = ai
    elif yy < ys:
        aa = aw
    else:
        aa = (ai+aw)/2
    a[i] = aa
return a

def myF(X):
    # Total number of unknowns is N+1
    N = len(X)-1
    T = X[0:N] # Be careful! Python list does not include last
    ys = X[N] # This is the N+1th element!

    # abar from eqn (14.5) of the notes
    abar = ai + (aw - ai)*ys*(1 - 0.241*(ys**2-1))
    # Tbar_inf from eqn (14.4) of the notes
    Tb = (Q*(1-abar) - A)/B

    # For the given T values, compute Phi at each y
    Phi = (k*Tb + Q*s(y)*(1 - afunc(y, ys)) - A)/(B+k)

    F = np.zeros(N+1)
    F[0:N] = T - Phi

    # Need one extra equation
    PhiC = (k*Tb + Q*s(ys)*(1 - (ai+aw)/2) - A)/(B+k)
    # Phi only has one entry
    # PhiC = PhiC[0]
    F[N] = PhiC - Tc

    return F

```

We should create an initial guess of the temperature profile. Here is a ‘random’ guess that goes from high to low.

```

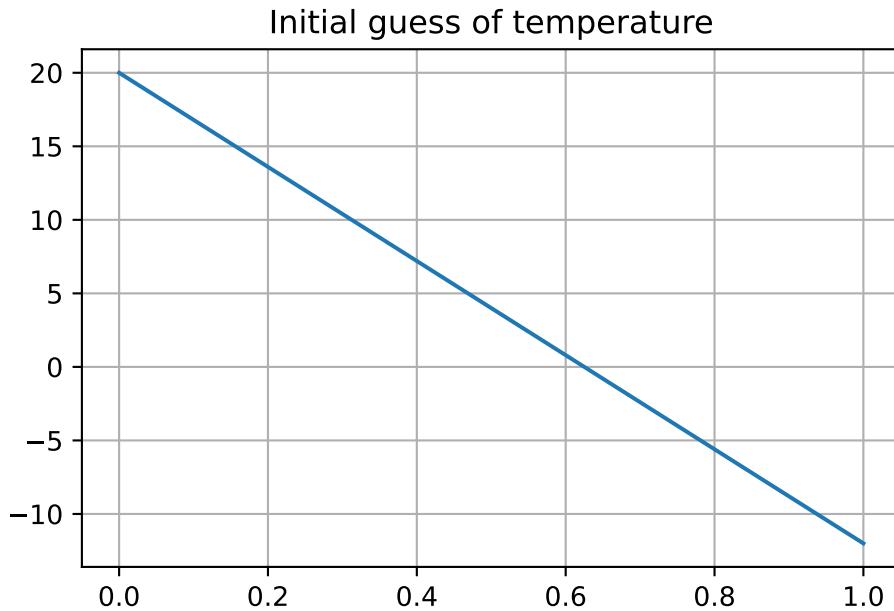
# Initialise the mesh with N points
N = 100
y = np.linspace(0, 1, N)

# Iterative scheme
# Form an initial guess
# Try this one for the fake solution
Tequator = 20; Tpole = -12;
T0 = Tequator + (Tpole - Tequator)*y

```

```
plt.plot(y, T0)
plt.grid(1)
plt.title("Initial guess of temperature")
```

```
Text(0.5, 1.0, 'Initial guess of temperature')
```



Now we attempt to solve the system twice, once with an initial guess of the ice line for small y and one with an initial guess with larger y . We should play around with parameters to make sure the solutions are robust, and also examine the residuals.

```
# We also need a guess of the ice line position
yguess = 0.3
# Form a system of N+1 unknowns
guess = np.append(T0, yguess)

# Run the solver
fwd = lambda X: myF(X)
sol, info, ier, msg = fsolve(fwd, guess, full_output=1)
print(msg)

T = sol[0:N]
ys1 = sol[N]
y = np.linspace(0, 1, N)

plt.plot(y, T, '-o')
plt.plot([ys1, ys1], [-20, 20], '--')
plt.grid(1)
plt.title("Ice line, ys1 = %1.3f" % ys1);

# Solve it again with a higher ice line position
yguess = 0.9
```

```

# Form a system of N+1 unknowns
guess = np.append(T0, ysguess)

# Run the solver
fwd = lambda X: myF(X)
sol, info, ier, msg = fsolve(fwd, guess, full_output=1)
print(msg)

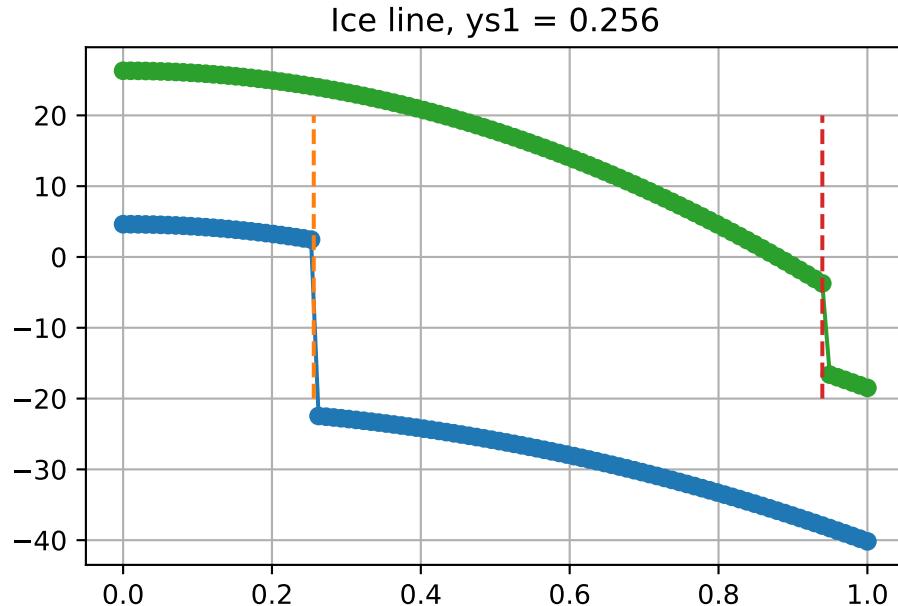
T = sol[0:N]
ys2 = sol[N]
y = np.linspace(0, 1, N)

plt.plot(y, T, '-o')
plt.plot([ys2, ys2], [-20, 20], '--')
plt.grid(1)
plt.title("Ice line, ys1 = %1.3f" % ys1);

```

The solution converged.

The solution converged.



Part IV

Fast-slow dynamics

This material will study the incidence of fast-flow dynamics that occur in generic dynamical systems. The emphasis will be on the posited fast-slow dynamics that occur in ice ages.

18

CHAPTER

Ice ages I: an introduction

Our next investigation will take us to the topic of fast-slow systems in order to posit toy mathematical models for ice ages.

18.1 Background

We have mainly been concerned about time periods in the decades, but it is now time to discuss much longer time periods. In particular, we are interested in the *Pleistocene era*, which lasts from around 2.5 million years ago to 11,700 years ago; this period marked the evolution of *Homo sapiens*. Begin by examining this figure:

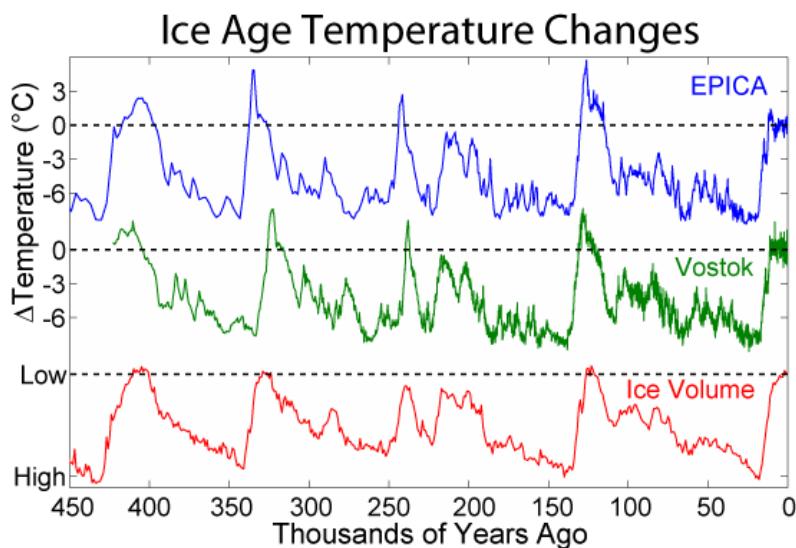
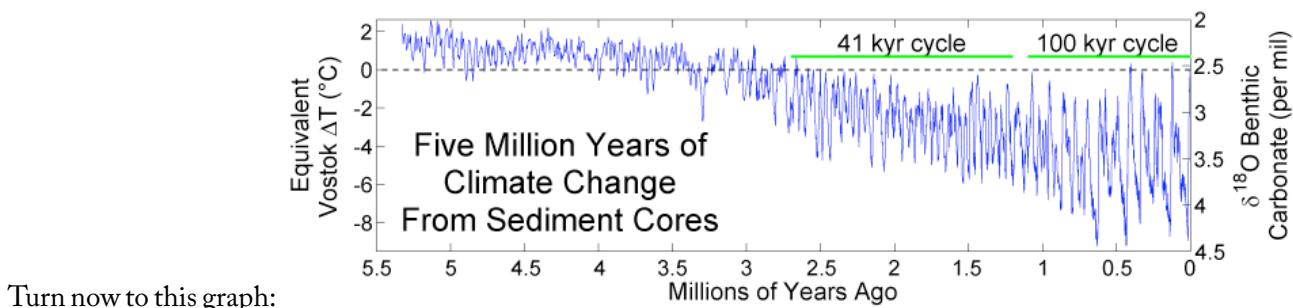


Figure 18.1: Figure from [here](#)

This figure indicates temperature changes tracked via various measures at different locations. The top two graphs show the temperatures derived from two different ice core measurements. The lower curve is the total ice cover. In this figure, you see evidence for the four complete glacial cycles with the Earth temperature dropping into the ice ages. A key observation concerns the regular periodic behaviour, with regular cycles appearing every 100k years.



Turn now to this graph:

This graph goes back even further. The key observation is that between 1 million and 2.5 million years, the periods seem to be predominantly 41k years. Thus sudden transition is referred to as the **Mid-Pleistocene Transition (MPT)**. The question of how or why the periods can suddenly change in such a manner is referred to as the **100,000-year problem**.

18.2 Milankovitch cycles

The most natural theory to explain the ice age periods is to consider that the Earth's orbit is affected by the gravitational attractions of the Sun, Moon, and the other planets. The complex interaction results in a slow, cyclic change in three important parameters of the orbit (Taylor 2005):

1. The *eccentricity* of the ellipse that the Earth describes in its orbit around the Sun each year (100kyr);
2. The angle between the equatorial plane and the orbital plane, known as the Earth's *obliquity* (41kyr);
3. The *precession* of the spin axis around the normal to the orbital plane (20kyr).

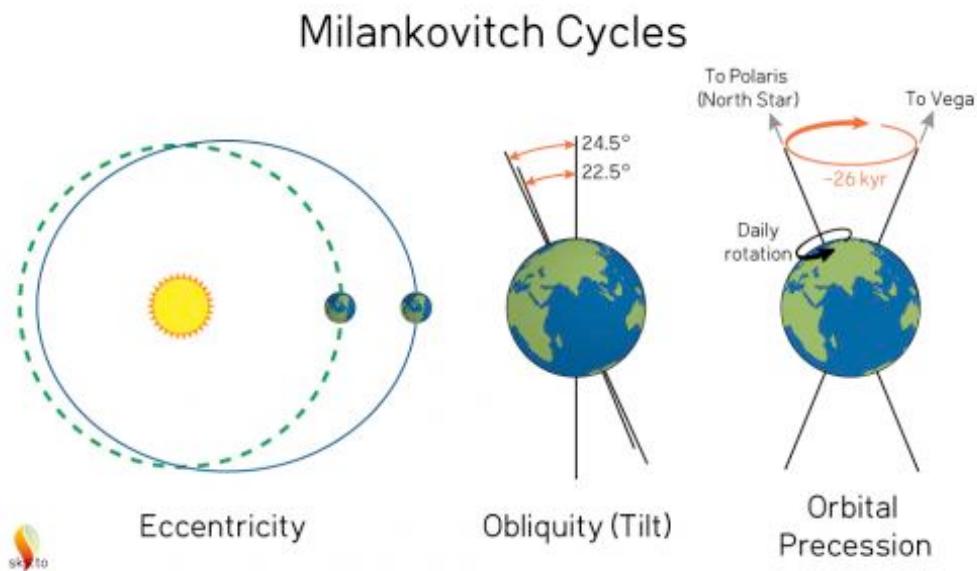


Figure 18.2: Figure from [here](#)

The name given to these cycles is *Milankovitch cycles*.

However, the key problem is that the Milankovich cycles do not seem to explain the MPT problem. The Milankovich cycles exhibit the necessary periods on both the 100kyr scale and 41kyr scale, but it cannot explain why there was a sudden transition. Moreover, the response of EBMs to the 100kyr forcing (due to the eccentricity) is not enough to explain why this forcing period takes precedence since it is the 41kyr cycle due to the obliquity that produces the largest effect on the solar radiation.

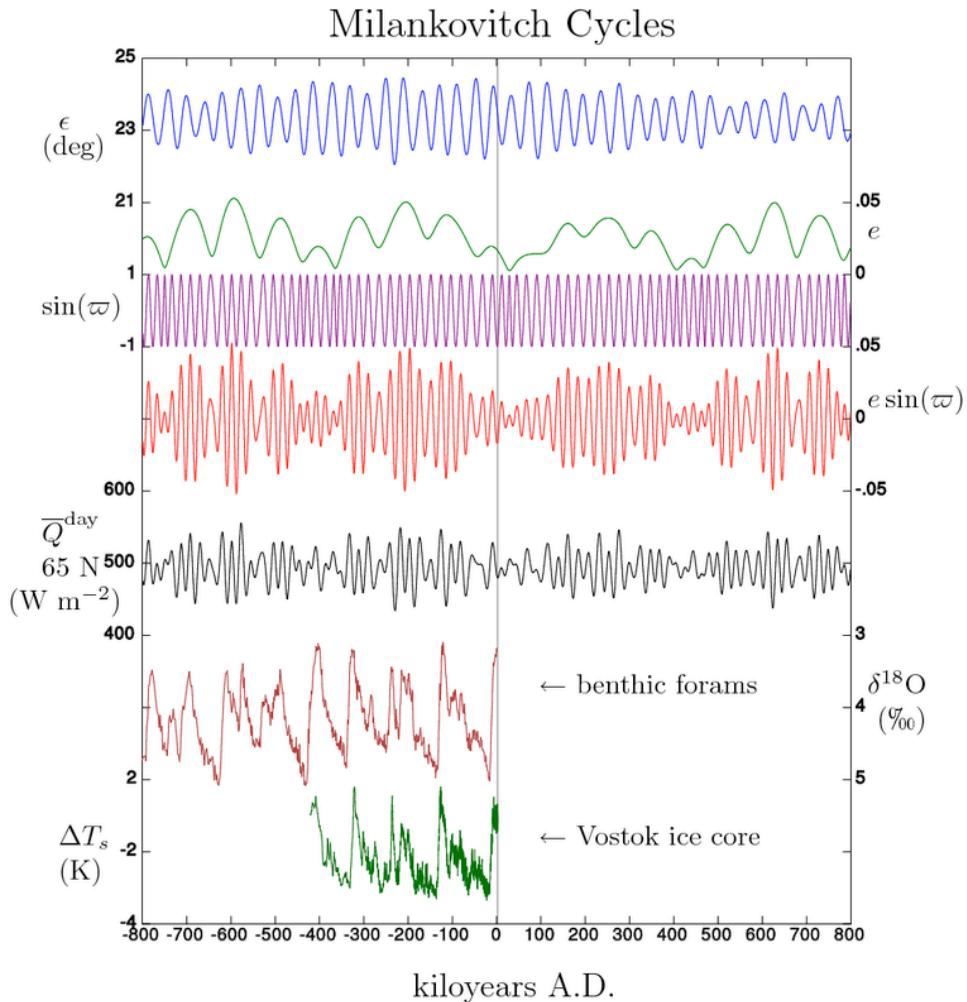


Figure 18.3: Figure from [here](#)

18.3 The mathematician's perspective on models

We start with some discussion of the perspective of a mathematician. The last few weeks, we have been studying models of Planet Earth that take the form of first-order autonomous systems:

$$C \frac{dT}{dt} = G(T),$$

or perhaps in the more recognisable ‘dynamical-systems’ form of

$$\dot{x} = f(x),$$

for some function $x = x(t)$. Because this is such a simple model, the analysis of the time-dependent evolution can be done on a **phase line**. Solutions can only exhibit two behaviours as $t \rightarrow \infty$: they can tend towards a fixed point or they can diverge (to $\pm\infty$).

There are numerous ways to extend the basic EBM. For instance:

1. **(Latitude variation)** Add latitude variation, with $G \mapsto G(y, T)$. Indeed this is what we had done in the last few chapters, and using a toy model (Newton's law of cooling) that imposes a driver $k(\bar{T} - T)$ into the ODE.
2. **(PDEs)** A more accurate model is to add a diffusion term, e.g. $k\nabla^2 T$. The Laplacian operator is then turned into a surface derivative in spherical coordinates along the planet. This turns the equation into a PDE, so is more involved to solve.
3. **(Non-autonomous forcing)** We can consider the addition of temporal forcing, for instance,

$$G \mapsto G(T) + \epsilon f(t).$$

This would model, for instance, some oscillatory source input. For instance, this might model extra oscillatory effects due to the influence of the other planets.

4. **(Systems of DEs)** We might be interested in extending the model so that, in addition to modelling the evolution of the temperature, we also model the evolution of CO₂ in the atmosphere. Or we might want a model where the albedo, itself, can evolve in time. An example of such a model is

$$\begin{aligned} C \frac{dT}{dt} &= Q[1 - a(t)] - \sigma\gamma T^4, \\ D \frac{da}{dt} &= a_{\text{eq}}(T) - a. \end{aligned}$$

The above provides a model for ice sheet dynamics.

18.4 Phase plane analysis for two-dimensional systems

Coming back now to the standard non-autonomous system of two ODEs, we have

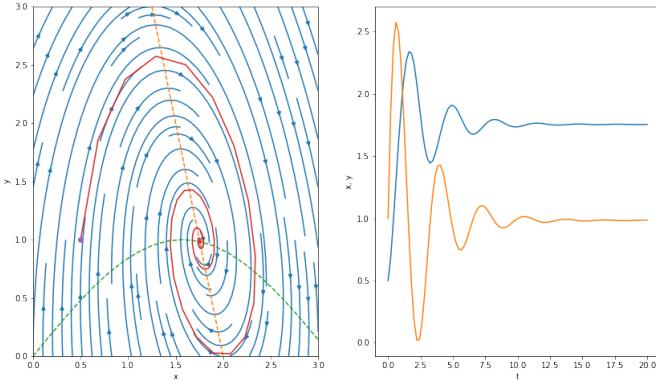
$$\begin{aligned} \dot{x} &= f_1(x, y), \\ \dot{y} &= f_2(x, y). \end{aligned}$$

What is the difference between these kinds of systems and the simpler 1D equations? The main difference is that the analysis is done in the **phase plane** given by (x, y) . Fixed points lie in the phase plane, but now there is the additional possibility of oscillations or circular orbits.

An example of this is the system

$$\begin{aligned} \dot{x} &= y - g(x), \\ \dot{y} &= h(x) - y. \end{aligned}$$

For instance, choosing $h(x) = 4(2 - x)$ and $g(x) = \sin x$ yields the following phase



plane diagram:

You can reproduce the phase plane via the code snippet in `lectures/Chap20-TypicalPhasePlot.ipynb`.

We see there exists a fixed point at where the two nullclines (dashed) intersect. Analysis near this fixed point would reveal that solutions encircle in a clockwise direction and tend towards the fixed point. This is shown by the solution plot on the right.

Therefore, one way to extend our EBMs is to consider coupling to another component. Keep this in mind as this is where we will be going in the next lecture.

19

CHAPTER

Ice ages II: a model for fast-slow dynamics

In order to capture the kind of dynamics observed in ice ages, we are interested in studying mathematical models that produce fast-slow dynamics – that is, dynamics that might consist of a slow evolution in one state, before rapidly jumping to another state; such dynamics might exhibit the kind of periodicity we observed in the profiles of temperatures through the ice ages. We might also be interested in understanding how the periodicity might suddenly change (the Mid-Pleistocene Transition problem) but this will be more challenging.

It turns out that such fast-slow systems are incredibly common. If we return to the canonical example presented in the previous chapter, we can modify it slightly so that it takes the form

$$\begin{aligned}\epsilon \dot{x} &= y - g(x), \\ \dot{y} &= h(x) - y.\end{aligned}$$

where ϵ is a small parameter.

Studying the above, notice that if ϵ is small, then we expect for $y \sim g(x)$, and therefore the dynamics should be constrained to lie along this given curve. However, this seems incompatible with the evolution that is specified by the second equation. The result is that this system will exhibit a sequence of slow dynamics (where it is near $y = g(x)$) followed by fast dynamics where it hops between the branches of the x -nullcline. This will be clear with an example.

19.1 The van der Pol oscillator

In an article by [Ditlevsen and Ashwin \(2018\)](#) the authors argued that the following van der Pol oscillator presents a good toy model for understanding some of the dynamics that occur in ice ages:

$$\frac{d^2x}{d\tau^2} = \frac{1}{\sqrt{\epsilon}}(1 - x^2)\frac{dx}{d\tau} - \alpha x + F(\tau),$$

where $x = x(t)$, $F(\tau) = A \cos(\omega_F \tau)$ and $\omega_F = (2\pi/41)\text{kyr}^{-1}$ represents a model for astronomical forcing. We will not discuss their ‘derivation’ of the above model (presented

in their sec. 5), though it seems best to consider it solely as a toy model since otherwise their derivation of its validity based on EBM arguments is somewhat sketchy.

In any case, we are firstly interested in the case of zero forcing, $F \equiv 0$. It can be verified that through a transformation of time, $\tau \mapsto t$, the above equation becomes

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

where we have chosen, without loss of generality, for $\alpha = 1$. Again we are considering $\epsilon \ll 1$. This is a very well-known problem in nonlinear oscillations and it is the canonical example of fast-slow dynamics.

The Lienard transformation

We first explain how to transform the above second-order equation so that it resembles the system of two linear differential equations shown above. Notice that the equation can be written in the form

$$\frac{d}{dt} \left(\epsilon \frac{dx}{dt} + \frac{1}{3}x^3 - x \right) + x = 0. \quad (19.1)$$

We then define

$$y \equiv \epsilon \frac{dx}{dt} + \frac{1}{3}x^3 - x$$

which can be re-arranged to yield

$$\epsilon \dot{x} = y - S(x), \quad S(x) \equiv \frac{1}{3}x^3 - x,$$

where we have used dots to indicate derivatives in time. The second equation is found by Equation 19.1 itself, since this yields

$$\dot{y} = -x.$$

Therefore together, we have our system:

$$\begin{aligned} \epsilon \dot{x} &= y - S(x), \\ \dot{y} &= -x. \end{aligned}$$

An investigation in Python

The lecture will now investigate the scripts to see the qualitative behaviour of the ODE. We will design a phase plane plotter in Python and see why this is called a fast-slow system.

The script can be found in the Noteable notes via `lectures/Chap21-DynamicsFastSlow.ipynb`.

During the lecture, we discussed the following image:

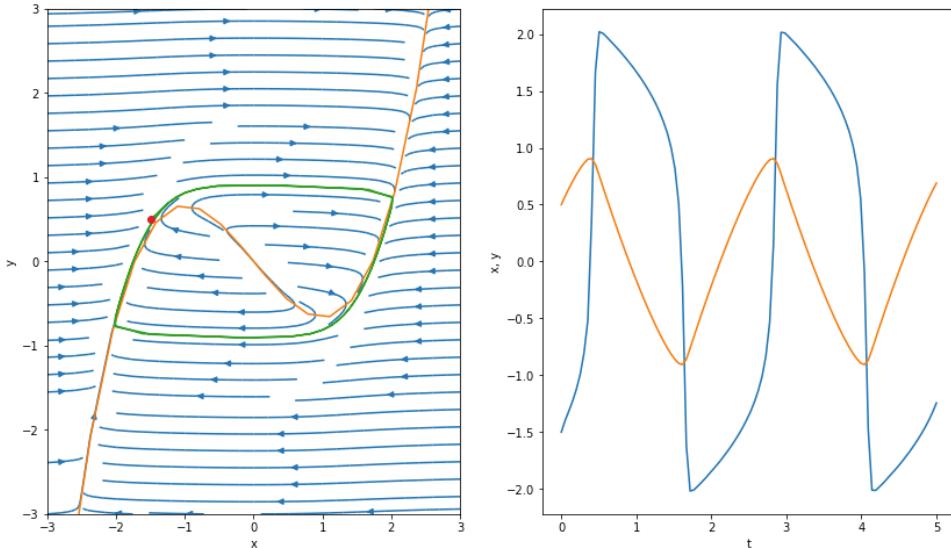


Figure 19.1: Phase plot of the above system

We interpreted the dynamics that follow either the slow manifold, $y = S(x)$, and also the fast transitions that occur between the branches of the cubic. Also shown on the right side of the above plots are the profiles, x and y versus t . In the limit $\epsilon \rightarrow 0$, the transitions begin to resemble shocks.

CHAPTER 20

Ice ages III: analysis of the van der Pol equation

In the previous chapter we introduced a canonical model for fast-slow dynamics given by the following system for $x(t)$ and $y(t)$:

$$\begin{aligned}\epsilon \dot{x} &= y - S(x), \\ \dot{y} &= -x.\end{aligned}$$

where $\epsilon > 0$ and $\epsilon \rightarrow 0$. This led to our study of then numerical computation of solutions and example graphics like this:

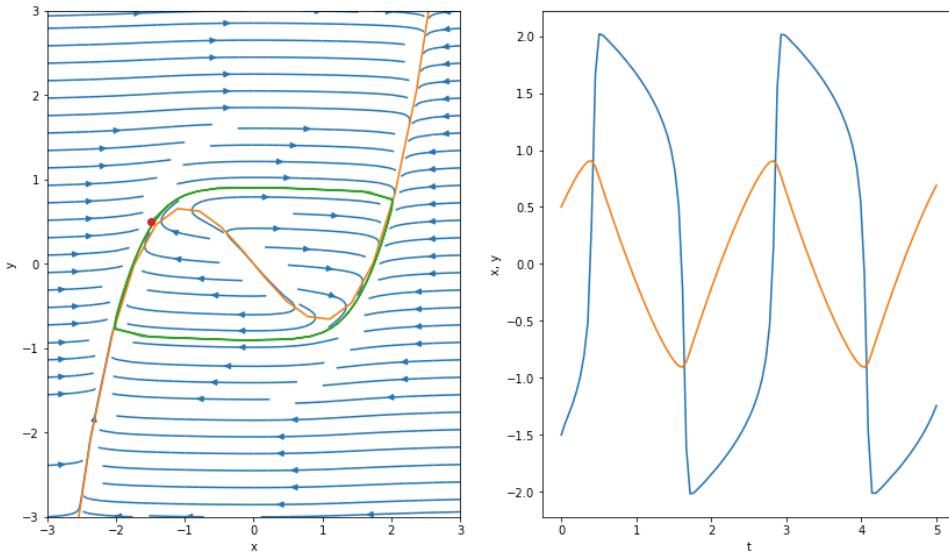


Figure 20.1: Phase plot of the above system

We now show how to perform the asymptotic analysis of the above system. The techniques used relate to the analysis of boundary layers studied in Chapter 8.

20.1 Analysis of the slow manifold

Let

$$x(t) = x_0(t) + \epsilon x_1(t) + \epsilon^2 x_2(t) + \dots, \quad y(t) = y_0(t) + \epsilon y_1(t) + \epsilon^2 y_2(t) + \dots$$

At leading order, we get the following system of equations:

$$\begin{aligned} 0 &= y_0 - S(x), \\ \dot{y}_0 &= -x_0. \end{aligned} \tag{20.1}$$

First, examine the differential equation on the second line. Since the rate-of-change of y is equal to $-x$, we can analyse the motion in the phase plane by the following cases:

$$y_0 \text{ is } \begin{cases} \text{decreasing} & \text{for } x_0 > 0 \\ \text{increasing} & \text{for } x_0 < 0 \end{cases}.$$

Therefore, we conclude that the y axis is nullcline in this leading-order approximation. The remaining equation restricts the solution to lie along the **slow manifold**,

$$y_0 = S(x) = \frac{1}{3}x_0^3 - x_0.$$

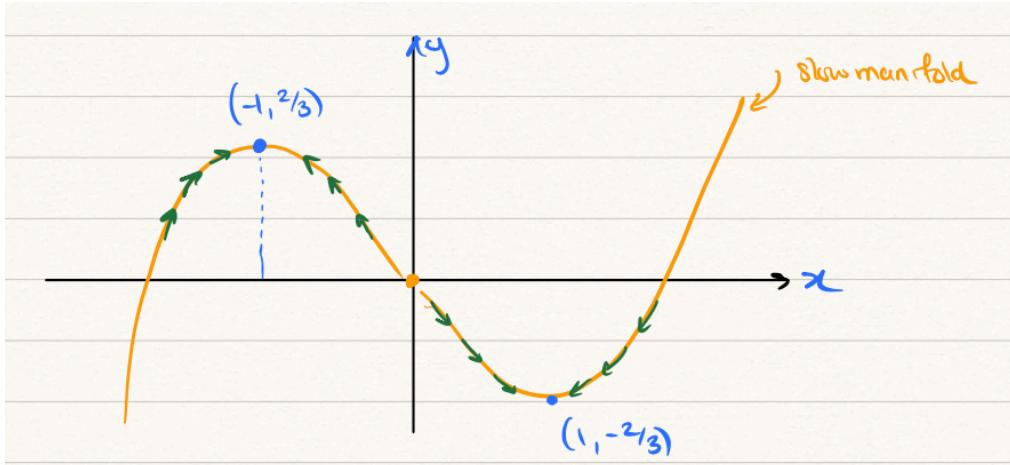


Figure 20.2: Illustration of dynamics along the slow manifold

According to our analysis, we should have both $y \sim S(x)$ as $\epsilon \rightarrow 0$ and also y increasing on $x < 0$ while decreasing on $x > 0$. But this yields an apparent contradiction as $x \rightarrow \pm 1$ and $y \rightarrow \mp 2/3$.

The issue is that, at the critical point(s) where $x \rightarrow \pm 1$, say $t = t^*$, the solution encounters a shock or boundary layer where the gradient \dot{x} is no longer small. Thus our prior assumption where we neglected $\epsilon \dot{x}$ is invalid.

20.2 Fast analysis

Near a shock, we re-scale

$$T = \frac{t - t^*}{\epsilon^\alpha} \implies t = t^* + \epsilon^\alpha T,$$

and set $x(t) = X(T)$ and $y(t) = Y(T)$ under this new coordinate change. Essentially, this re-scales the coordinate axes, originally in t , near the unknown location of the shock, $t = t^*$. We can verify that the correct balance for re-scaling is chosen with $\alpha = 1$. From the chain rule, we convert derivatives as follows:

$$\frac{d}{dt} = \frac{dT}{dt} \frac{d}{dT} = \frac{1}{\epsilon} \frac{d}{dT}.$$

Then the system becomes

$$\begin{aligned}\frac{dX}{dT} &= Y - S(X) \\ \frac{dY}{dT} &= -\epsilon X.\end{aligned}$$

We expand now in the typical fashion, setting

$$\begin{aligned}X &= X_0(T) + \epsilon X_1(T) + \dots \\ Y &= Y_0(T) + \epsilon Y_1(T) + \dots\end{aligned}$$

At leading order, this now yields

$$\begin{aligned}\frac{dX_0}{dT} &= Y_0 - S(X_0) \\ \frac{dY_0}{dT} &= 0.\end{aligned}$$

Therefore, from the second equation, we obtain

$$Y_0 = \text{constant}$$

which is to be expected from the phase-plane diagrams we have shown. Since we have established that the max/min of the slow manifold, $S(X)$, is given by $\pm 2/3$, then we know that the above constant is this value *if the solution had begun from the slow manifold* (see comment later about the initial condition).

Next, the equation for X_0 has the form of a first-order nonlinear differential equation that can be studied by phase-line analysis. If the constant Y_0 is indeed $\pm 2/3$, then the differential equation for the other component takes the form

$$\frac{dX_0}{dT} = Y_0 - S(X_0) = \pm \frac{2}{3} - S(X_0).$$

This leads to the following picture.

To begin, we consider the evolution from the points marked ‘A’ and ‘B’. This occurs along the slow manifold. At the point ‘B’, time reaches the critical point, $t \rightarrow t^*$, and the solutions enter the fast dynamics, following along points ‘B’ to ‘C’. Notice that along this segment in the phase plane,

$$\frac{2}{3} - S(X_0) > 0,$$

therefore X_0 must increase during this time. However, upon approaching the point ‘C’, the above LHS tends to zero, and hence X_0 slows its increase. This exists the fast dynamics layer and again the solution continues along the slow manifold from ‘C’ to ‘D’. Upon reaching ‘D’, it again enters a stage of fast dynamics (the t^* point here is a new one).

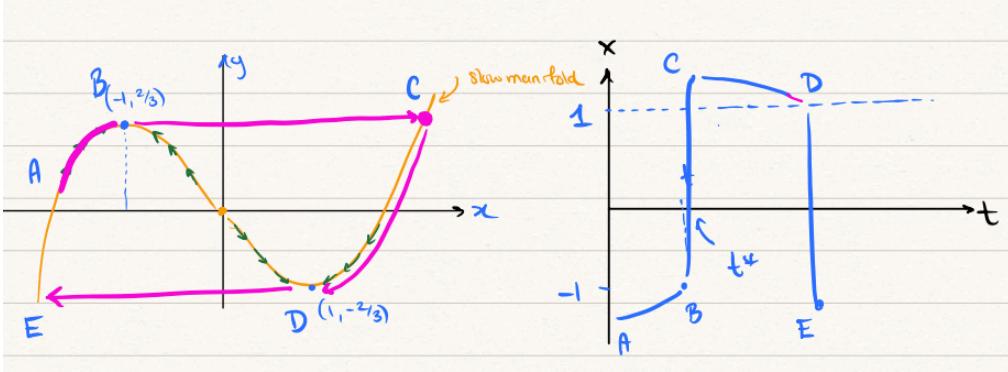


Figure 20.3: Fast dynamics

20.3 Initial conditions

You will notice that above, we have not really discussed the effect of the initial conditions, say,

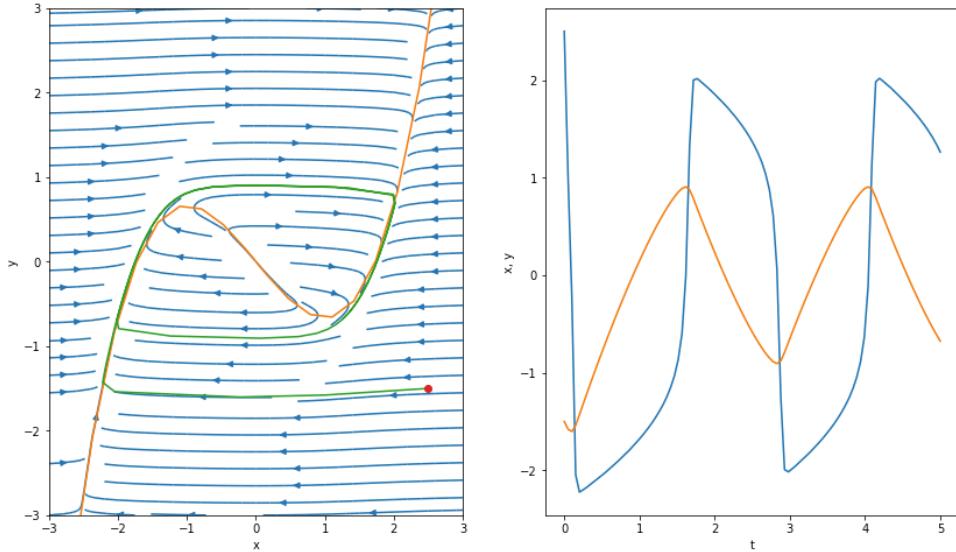
$$\begin{aligned} x(0) &= A, \\ y(0) &= B. \end{aligned}$$

Perhaps more specifically, we have performed our asymptotic analysis with the assumption that the solution starts directly on the slow manifold, i.e. $y \sim S(x)$. Returning to the slow analysis leading to Equation 20.1 notice that the above conditions will not in general be compatible with the equations (since 'A' and 'B' might be chosen completely different from those values where $y = S(x)$). The conclusion is then that the slow analysis applied cannot be valid near the initial condition.

This turns out to be very similar to the example studied in Chapter 8, which also had a time-dependent problem where the initial condition was incompatible.

Therefore near $t = 0$, we must introduce an analogous boundary layer. This analysis will be identical to the above fast-scale dynamics except that $t^* = 0$. The conclusion is that if we begin off the manifold $y = S(x)$, the solution will rapidly tend left or right in the phase plane (at near-constant values of y). Once it intersects with the manifold, the trajectories then resume the slow and fast dynamics observed above.

This is now shown below for the initial condition of $x(0) = 2.5$, $y(0) = -1.5$ and $\epsilon = 0.05$.



Notice the red circle in the left phase plot showing the initial condition. We see that the solutions then exhibit the fast dynamics, with x decreasing, until encountering the slow manifold. Thereafter it follows the standard slow-fast dynamics.

On the right image, notice the significant shock or boundary layer near $t = 0$. If desired, we could confirm the size of this boundary-layer regions scales with ϵ .

CHAPTER 21

Fast-slow dynamics for higher-order systems

In the last two lectures, we discussed the use of toy models that exhibit fast-slow dynamics. These have been used to posit possible changes in behaviour in ice ages, where historical records confirm that multiple periodic states are possible.

It is important to note that the van der Pol oscillator we have studied, which is equivalent to two first-order differential equations, is really not that unusual, and systems of (three or more) differential equations will often exhibit the kind of fast-slow behaviours observed. This occurs commonly in applications much more general than climate or geoscience—of course, if your view is to concentrate to models of Planet Earth, you can expect this kind of behaviour whenever you have moderately complicated systems of coupled oscillators describing the planet!

21.1 A chemical model

As a means of providing you with additional toy examples, let us examine a model used in studying chemical reactions. Consider an enzyme-mediated chemical reaction given by the following: a substrate (S) and enzyme (E) combine to produce a complex (C) with a reaction rate k_1 . The complex then produces a product (P) and enzyme (E) with reaction rate k_3 . Finally the complex (C) can degrade into (S) and (E) with rate k_2 . The enzyme E is a compound that allows for an immediate reaction step that leads to the overall reaction, which in this case is $S \rightarrow P$. The intermediate substance is C , the complex. The point here is that dependent on the rate constants, k_1 , k_2 , k_3 , and the amount of each substance, the overall chemical reaction may be coarse-grained so that you only observe $S \rightarrow P$ (this is the slow dynamics).

We can consider the dynamics of the three substances and derive the following system of three equations

$$\begin{aligned} \frac{ds}{dt} &= -s(1 - c) + \lambda c, & s(0) &= 1, \\ \epsilon \frac{dc}{dt} &= s(1 - c) - \mu c, & c(0) &= 0, \\ \frac{dp}{dt} &= (\mu - \lambda)c, & p(0) &= 0. \end{aligned} \tag{21.1}$$

where the three substances have been suitably non-dimensionalised. The key is to assume that the parameters $\lambda = O(1) = \mu$, while

$$\epsilon = \frac{\text{initial enzyme}}{\text{initial substrate}} \ll 1.$$

Here is a typical numerical calculation found in the script `lecture28-ThreeTermChemicalReaction.ipynb`.

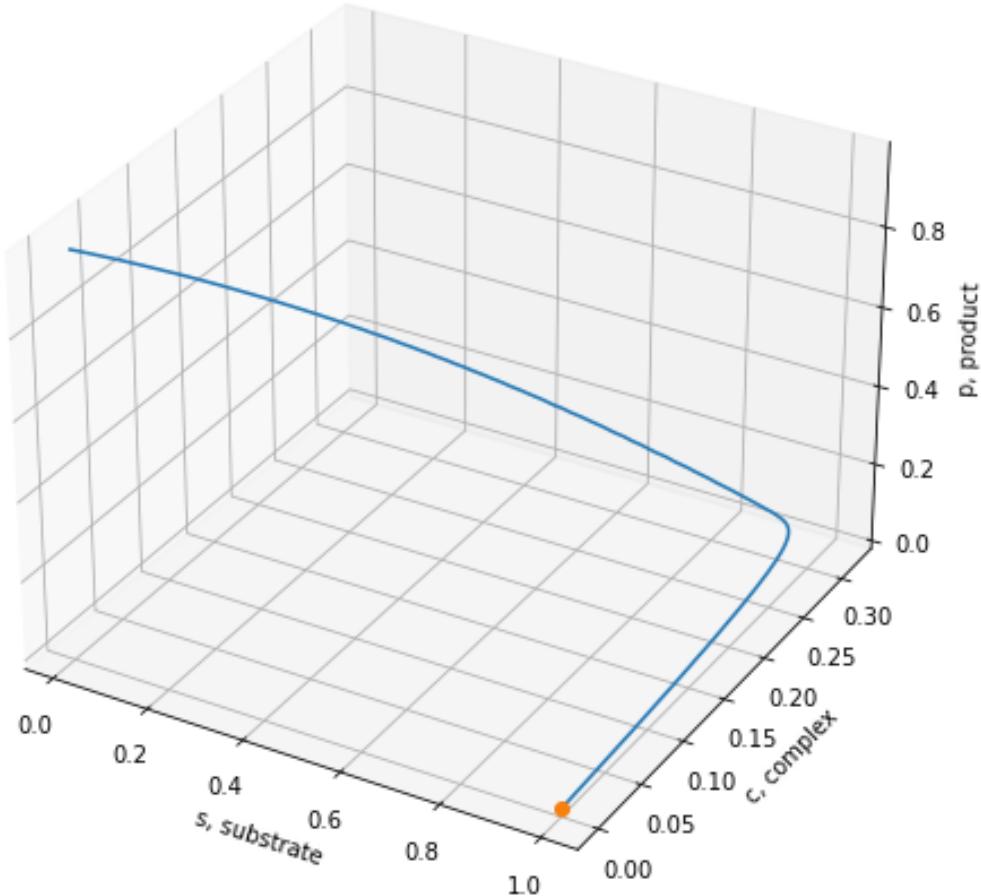


Figure 21.1: Numerical calculation of the system of three equations with $\lambda = 1$, $\mu = 2$, and $\epsilon = 0.1$.

Here is the same solution but now plotted as a function of time.

The above graph suggests that there is a boundary layer for the complex near $t = 0$.

21.2 Late-time behaviour

Note that as $t \rightarrow \infty$, all the derivative terms are set to zero. This tells us that

$$s \rightarrow 0, qquad c \rightarrow 0.$$

Note that it is not immediately obvious what the limit of p is. But it can be shown (see below) that $p \rightarrow 1$.

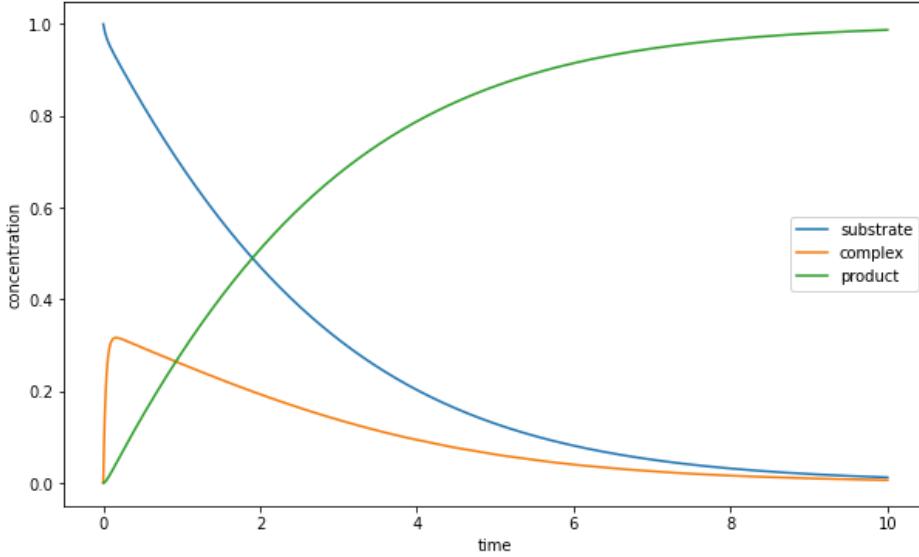


Figure 21.2: Plot of the solution evolution with time

21.3 Slow analysis

We thus expand the solutions in the typical way, setting

$$\begin{aligned}s(t) &= s_0(t) + \epsilon s_1(t) + \dots \\ c(t) &= c_0(t) + \epsilon c_1(t) + \dots \\ p(t) &= p_0(t) + \epsilon p_1(t) + \dots\end{aligned}$$

This yields at leading order,

$$\begin{aligned}\frac{ds_0}{dt} &= -s_0(1 - c_0) + \lambda c_0, \\ 0 &= s_0(1 - c_0) - \mu c_0, \\ \frac{dp_0}{dt} &= (\mu - \lambda)c_0,\end{aligned}\tag{21.2}$$

The second equation yields the slow manifold, for which we expect

$$s(c_0) = \frac{\mu c_0}{1 - c_0}.\tag{21.3}$$

21.4 Fast dynamics near the initial condition

As with our van der Pol example, the initial condition, where $c(0) = 0$ and $s(0) = 1$, does not lie on the slow manifold above. Therefore we expect that there is a boundary layer where the system rapidly begins from the initial condition and tends to the slow manifold. The distinguished scaling on time can be confirmed to be

$$t = \epsilon^\alpha T = \epsilon T,$$

where $\alpha = 1$. We then set

$$s = S(T) \quad c = C(T),$$

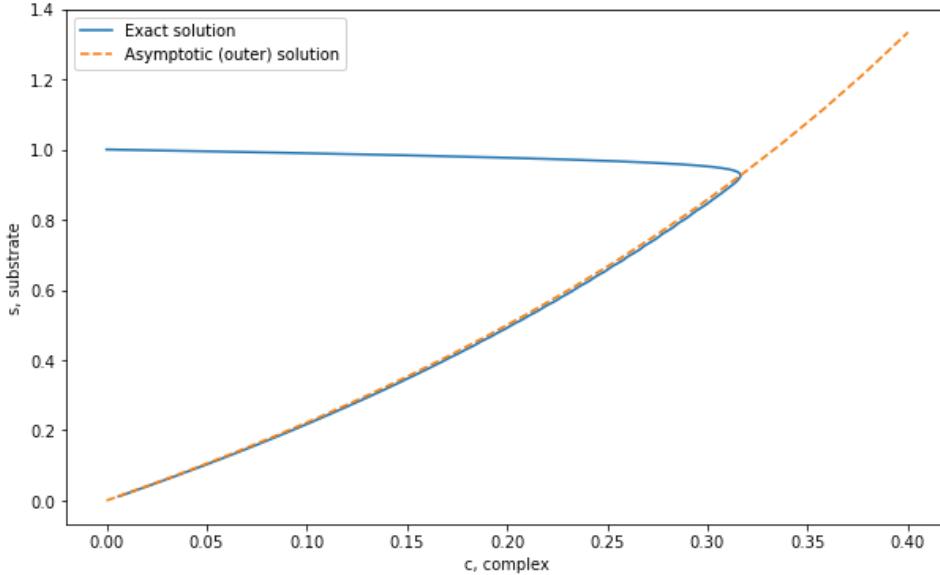


Figure 21.3: Numerical solution as before but now plotted in the (c, s) phase plane. The dashed line is the approximation derived above.

for our new unknowns within this later. Under the scaling, the first two ODEs in (Equation 21.1) become

$$\begin{aligned} \frac{dS_0}{dT} &= 0, & S_0(0) &= 1 \\ \frac{dC_0}{dT} &= S_0(1 - C_0) - \mu C_0. & C_0 &= 0. \end{aligned}$$

Therefore, you see that the leading-order value of S is expected to be constant, and it remains at its initial value,

$$S_0(T) \equiv 1.$$

Although we could solve for the concentration of c or C , this turns out not to be necessary. It should first be remarked that as the outer solution, $s \sim s_0$, tends to the initial condition, $t \rightarrow 0$, we should have

$$\lim_{t \rightarrow 0} s_0(t) = 1 = \lim_{T \rightarrow \infty} S_0(T).$$

Compare this with the condition (Equation 8.2) when we initially studied the method of matched asymptotics. Therefore, even though there is a boundary layer necessary for the concentration c near $t = 0$, we do not need to resolve it.

21.5 An equation for the substrate and product

Take now (Equation 21.3), which yields

$$c_0 = \frac{s_0}{\mu + s_0},$$

and substitute this into the first and third equations (Equation 21.2). This gives

$$\begin{aligned} \frac{ds_0}{dt} &= -\frac{\mu - \lambda}{\mu + s_0} s_0, \\ \frac{dp_0}{dt} &= \frac{\mu - \lambda}{\mu + s_0} s_0 \end{aligned}$$

Notice that actually, if we add these two equations, we come out with

$$\frac{d}{dt}(s_0 + p_0) = 0,$$

which is a conservation of mass/concentration statement that indicates that the total substrate and product is conserved. Therefore, if we only solve the first equation for s_0 :

$$\frac{ds_0}{dt} = -\frac{\mu - \lambda}{\mu + s_0} s_0, \quad s_0(0) = 1, \quad (21.4)$$

we can obtain the product concentration by $p_0 = 1 - s_0$.

The above equation (Equation 21.4) is in fact a famous result in biochemistry. It indicates that the substrate concentration follows a nonlinear rate and is called *the Michaelis-Menten law*. We have significantly simplified the necessary mathematical analysis down from the analysis of coupled system of three equations to a single equation for a single unknown (s_0)!

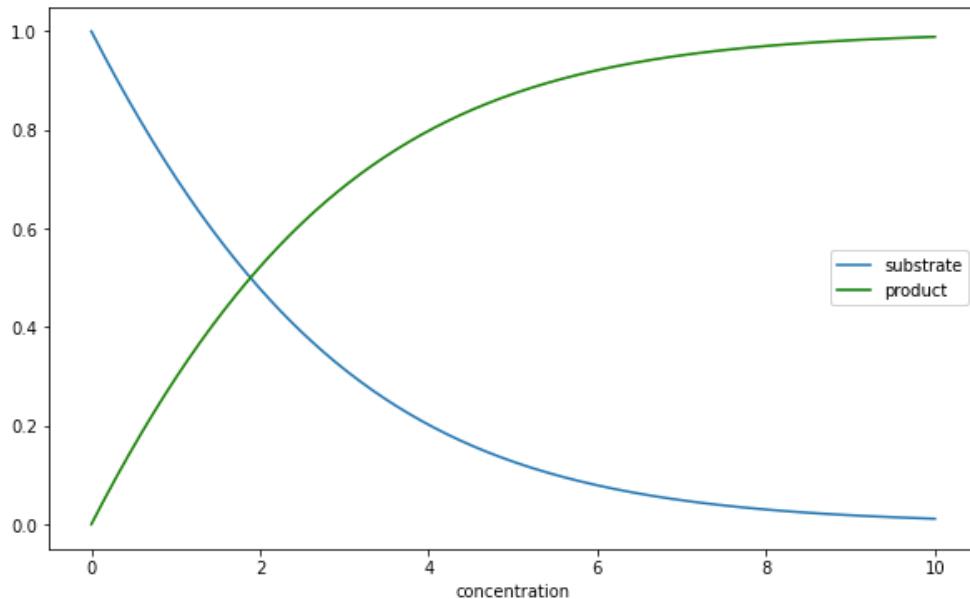


Figure 21.4: Numerical solution of the single differential equation

Part V

Box models of the ocean and environment

CHAPTER 22

Basic models of the ocean

22.1 Terminology and context

The ocean plays a significant role in regulating the Earth's climate, as it acts as a massive heat sink and helps to distribute heat and moisture around the planet. In addition, carbon dioxide is water soluble, and through precipitation and wave motion, is transferred into the oceans. Thus the ocean acts as a sink, absorbing large amounts of this greenhouse gas from the atmosphere.

The thermohaline circulation (THC), also known as the global ocean conveyor belt, is a complex ocean circulation pattern that is driven by differences in water temperature and salinity. It is an important component of the Earth's climate system, as it helps to distribute heat and other properties throughout the planet's oceans.

The thermohaline circulation is driven by the sinking of cold, dense water in the polar regions, which then spreads out and flows towards the equator. As the water warms and becomes less dense, it rises to the surface and returns to the poles, creating a continuous loop of ocean currents. The role of salinity in driving the thermohaline circulation is due to the fact that the dissolved salts in seawater increase its density.

This circulation pattern has a significant impact on global climate, as it helps to regulate the exchange of heat and other properties between the oceans and the atmosphere. Changes in the thermohaline circulation, such as those caused by global warming, can have far-reaching effects on the planet's climate and weather patterns.

22.2 Temperature

Extra content

The material in this subsection is not examinable and is given here for further context. It can be safely ignored.

In regards to the temperature, the ocean can be divided into three layers.

1. The top layer is thin (on the order of metres) and is heated from the Sun. Mixing is a dominant effect due to wind and waves, and so the temperature in this region is mostly constant.

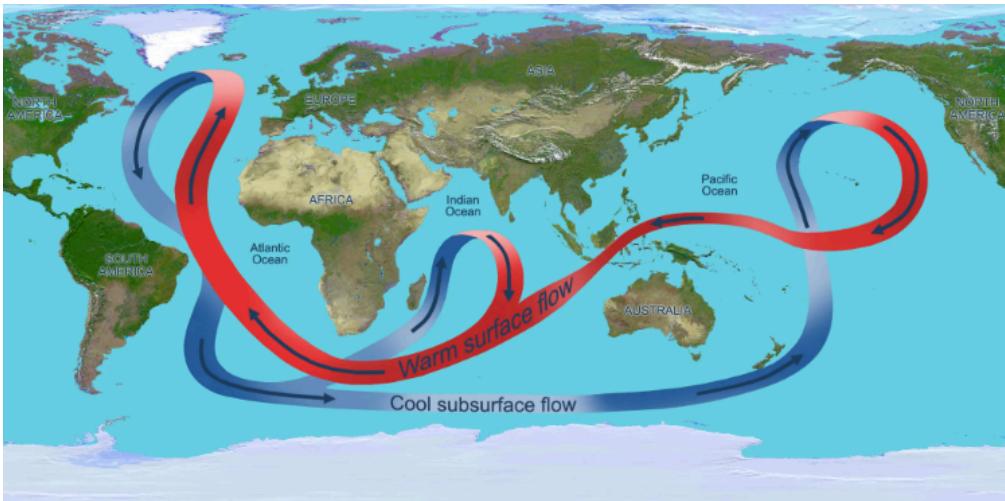


Figure 22.1: The thermohaline circulation. In the Atlantic, the circulation carries warm water (red) north near the surface and cold deep water (blue) south. Image from NASA/JPL.

2. The *thermocline region* is the intermediate layer. Here, the temperature decreases approximately linearly.
3. The deep *abyssal zone* comprises 98% of the total volume of the oceans. The temperature in this region is mostly constant, and a few degrees above freezing.

Within the intermediate region, the temperature can be modelled by an advection diffusion equation,

$$\frac{\partial T}{\partial t} + w \frac{\partial T}{\partial z} = \kappa \frac{\partial^2 T}{\partial z^2}, \quad (22.1)$$

where w is the upswelling velocity and κ is the diffusion coefficient of the fluid.

Let us assume that the temperature in this region is near steady state and that the upwards velocity is constant. Then we integrate the ODE to find

$$T(z) = T_0 + T_1 e^{-z/z^*},$$

where T_0 and T_1 are constants. From (Kaper und Engler, 2013), the typical orders for the parameters are $\kappa \sim 10^{-2} \text{ m}^2 \text{ s}^{-1}$ and $z^* \sim 10^2 \text{ m}$, so $w \sim 10^{-4} \text{ m s}^{-1}$, which is quite slow.

22.3 Salinity and density

Extra content

The material in this subsection is not examinable and is given here for further context. It can be safely ignored.

Salinity is a key component in the oceans since the salts have a large effect on the water density (which consequently drives motion). Salinity is measured in psu or *practical salinity units*, which is a non-dimensional ratio of conductivities. In the mixed layer, the salinity ranges from 31–39 psu, and is about 35 in the abyssal zone. You can inspect the profile in Figure 22.2.

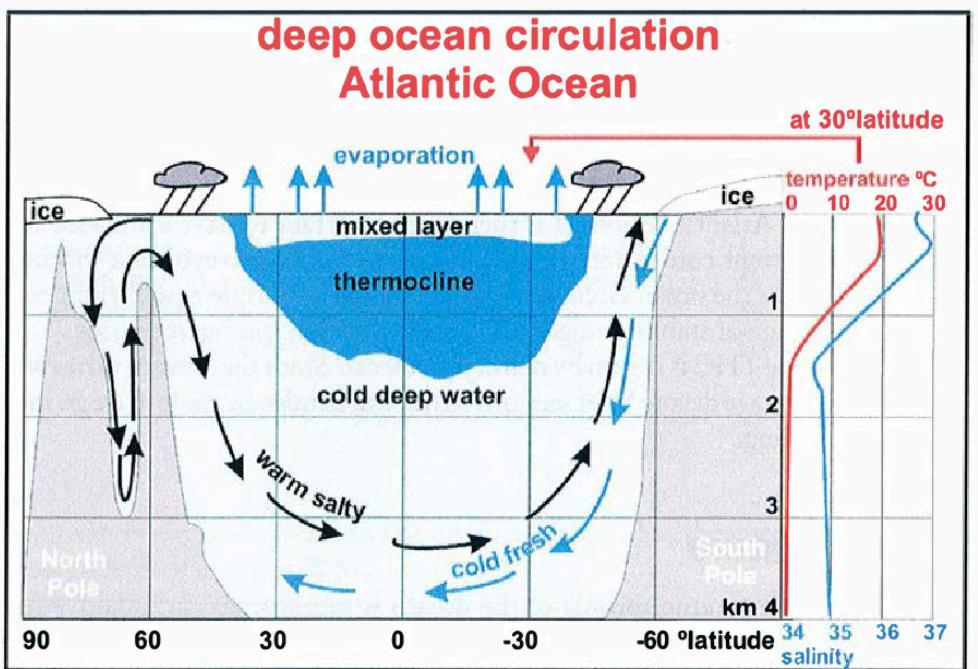


Figure 3.4. Cross-section of the Atlantic Ocean. Reprinted with permission from Seafriends.

Figure 22.2: Cross section of the Atlantic Ocean, showing the temperature and salinity profiles on the right. Image from (Kaper und Engler, 2013).

Below in the study of the ocean, we will need a so-called *equation of state*, which connects the density, ρ , with the salinity, say S , and temperature, say T . Here is a typical graph of the relationship.

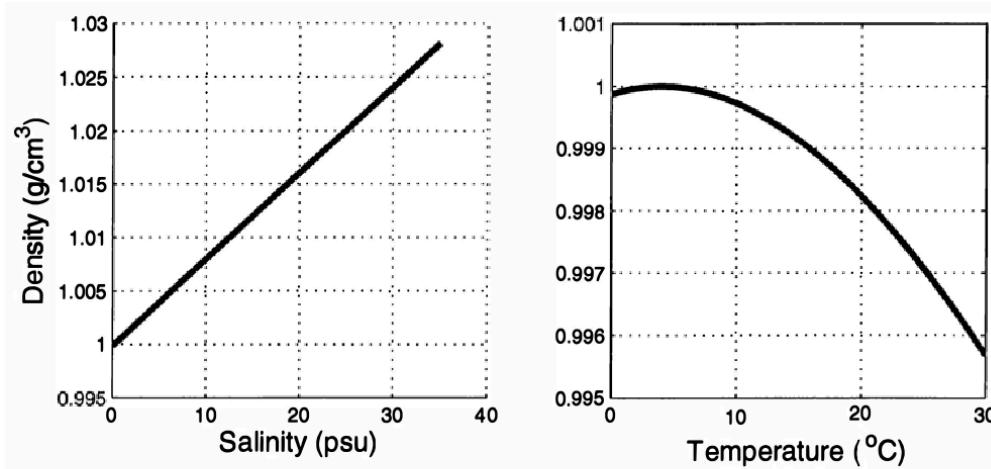


Figure 3.5. Density as a function of salinity and temperature.

Figure 22.3: Image from (Kaper und Engler, 2013)

22.4 Two-box model of the ocean

Modelling the THC is a challenging task! In principle, this might involve the solution of multiple coupled PDEs for the flows and temperatures, which would then need to

be solved on a very complicated domain. In addition, such models require a number of empirical equations of state (connecting density to temperature and salinity).

Toy models can be developed much more easily at the ‘systems level’ via box models.

22.5 A one-dimensional model (constant temperature)

Construction and assumptions of the box model.

1. We consider two boxes, labeled ‘1’ and ‘2’, respectively via subscripts. Box 1 corresponds to high latitudes (near poles) and Box 2 to low latitudes (near equator). Each box has a corresponding temperature, T_i , and salinity, S_i . In each box, the temperature and salinities are well mixed, so they take different values in either box, in general.

Differences between the two boxes will drive a flow through the capillary pipe that connects the boxes at the bottom. A compensating flow at the surface ensures that the volume of water in each box does not change. Together, these two exchanges (top and bottom) represent the overturning circulation of the THC.

2. We assume (see above) that the strength of the exchange flow between the boxes is linearly proportional to their differences of temperature and salinity. In particular, let ρ_1 and ρ_2 be the two densities in the corresponding boxes. The flow q in the capillary pipe is driven by the difference

$$q = k \frac{\rho_1 - \rho_2}{\rho_0}, \quad (22.2)$$

where k is the hydraulic constant and takes the typical value of $k = 1.5 \cdot 10^{-6} \text{ s}^{-1}$. The reference density ρ_0 is defined in conjunction with the equation of state.

In the normal state of the planet, $q > 0$ when the flow through the capillary goes in the direction of the equator as a result of higher densities at high latitudes.

3. We need an equation of state that connects the densities, ρ , with the temperature and salinities. This concerns the figure Figure 22.3. We assume that

$$\rho = \rho_0(1 - \alpha(T - T_0) + \beta(S - S_0)), \quad (22.3)$$

where T_0 and S_0 are average values of temperature and salinity; ρ_0 is the density if $T = T_0$ and $S = S_0$; α and β are constants with typical values $\alpha = 1.5 \times 10^{-4} \text{ deg}^{-1}$ and $\beta = 8 \times 10^{-4} \text{ psu}^{-1}$. Combining (Equation 22.2) with (Equation 22.3) gives

$$q = k[\alpha(T_2 - T_1) - \beta(S_2 - S_1)]. \quad (22.4)$$

4. External wind forces and Coriolis effects are ignored.
5. We assume that in each box, there is an exchange of heat and salinity to the surrounding environment. For instance, salinity will exchange due to evaporation, precipitation, and runoff. These will have constant of proportionality of c and d in the equations below.
6. Precipitation, evaporation, and runoff from the continents and atmosphere will also cause the salinity in either box to change. We therefore assume that this is modeled by a flux H .

Remember that in general, $T_2 > T_1$ (the temperatures at the equator are higher). Consequently, evaporation dominates Box 2; there is a net loss of salt-free moisture in the atmosphere and this causes a compensating virtual flux of salt, $H > 0$, into Box 1. At the same time, there is higher precipitation and runoff into Box 1 (due to lower temperatures), causing a net gain of salt-free moisture and therefore a loss of virtual salt, say $-H < 0$. We assume these two virtual salt fluxes sum to zero.

Note that in the normal state of affairs, $S_2 > S_1$, and the lower latitude water is saltier than that of high latitudes.

An image of the box model is shown below.

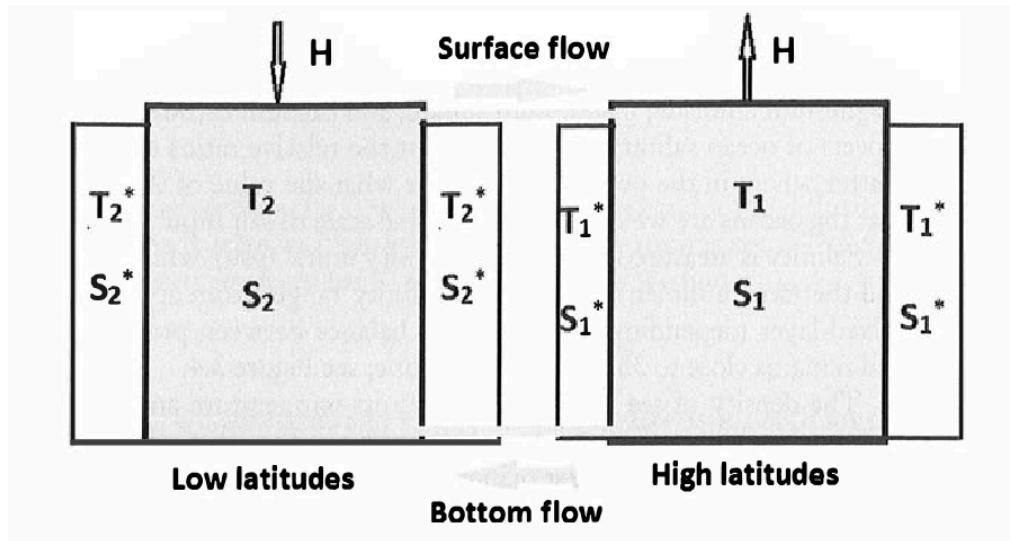


Figure 22.4: Two-box model of the North Atlantic with evaporation and precipitation.
Image from (Kaper und Engler, 2013)

The equations are given as follows.

$$\begin{aligned}\frac{dT_1}{dt} &= c(T_1^* - T_1) + |q|(T_2 - T_1) \\ \frac{dT_2}{dt} &= c(T_2^* - T_2) + |q|(T_1 - T_2) \\ \frac{dS_1}{dt} &= -H + d(S_1^* - S_1) + |q|(S_2 - S_1) \\ \frac{dS_2}{dt} &= H + d(S_2^* - S_2) + |q|(S_1 - S_2)\end{aligned}\tag{22.5}$$

Note that in the exchange terms corresponding to the flow q in the capillary pipes, we use an absolute value since it does not matter (flows are exchanged in both lower and upper pipes).

Note also that $q = q(T_1, T_2, S_1, S_2)$ as given in (Equation 22.4).

22.6 Reducing the equations via symmetry

Extra content

The manipulations in this subsection are, unfortunately, necessary but are not examinable. You simply should accept the final set of equations as the baseline. In essence, it is possible to shift the temperatures and salinities so as to eliminate two constants from the model. Unfortunately, the complexity of such models means you have to do a lot to ‘prepare’ them for analysis.

By adding the equations (Equation 22.5), it can be verified that in the limit $t \rightarrow \infty$, the average temperature in the system, $1/2(T_1 + T_2)$ tends to the average temperature in the surrounding basins $1/2(T_1^* + T_2^*)$. The same conclusion can be drawn for the corresponding salinities.

It is then sensible to write all temperature and salinity in terms of this baseline scenario. So let us write

$$\begin{aligned} T_1 &= \frac{1}{2}m + U_1, \\ T_2 &= \frac{1}{2}m + U_2, \end{aligned}$$

where $m = T_1^* + T_2^*$. Then the first equation becomes

$$\begin{aligned} \frac{dU_1}{dt} &= c \left(T_1^* - U_1 - \frac{1}{2}m \right) + |q|(U_2 - U_1) \\ &= c(-T^* - U_1) + |q|(U_2 - U_1). \end{aligned}$$

where $T^* = \frac{1}{2}(T_2^* - T_1^*)$.

The analogous manipulations are done to the quantities for the salinity. In the end, if we (confusingly) re-write T for U , then we obtain

$$\begin{aligned} \frac{dT_1}{dt} &= c(-T^* - T_1) + |q|(T_2 - T_1) \\ \frac{dT_2}{dt} &= c(T^* - T_2) + |q|(T_1 - T_2) \\ \frac{dS_1}{dt} &= -H + d(-S^* - S_1) + |q|(S_2 - S_1) \\ \frac{dS_2}{dt} &= H + d(S^* - S_2) + |q|(S_1 - S_2) \end{aligned} \tag{22.6}$$

Comparing the above to Equation 22.5, the main difference is that, in expressing the temperature and salinities with respect to the average values in the basin, we have eliminated two of the constants from the set $(T_1^*, T_2^*, S_1^*, S_2^*)$ now only into two constants (T^*, S^*) .

In the situation of zero salt flux. $H = 0$, the above model reduces to Stommel’s box model studied later in this course.

22.7 Analysis of a reduced 1D model for the salinity

It is possible to do some analysis on the system (Equation 22.6), though we may prefer to simply move on to study the more historically well-known reduction called *Stommel’s box model*.

We make the following assumptions:

- We assume that on the timescale of interest in the THC, the temperature of each box equilibrates quickly with the surrounding basin. Therefore the system can be regarded as being in steady state.
- The difference in temperatures between the two boxes is small; together with the top assumption, this implies that $T_1(t) = -T^*$ and $T_2(t) = T^*$.
- Salinity exchanges by negligible amounts with its surrounding basin, i.e. $d = 0$

This leaves us with

$$\begin{aligned}\frac{dS_1}{dt} &= -H + |q|(S_2 - S_1) \\ \frac{dS_2}{dt} &= H + |q|(S_1 - S_2),\end{aligned}$$

where $q = k(2\alpha T^* - \beta(S_2 - S_1))$.

Now, the formulation for the salinities can be placed into a single equation for $\Delta S = S_2 - S_1$, which satisfies

$$\frac{d\Delta S}{dt} = 2H - 2k|\alpha\Delta T - \beta\Delta S|\Delta S. \quad (22.7)$$

This is now a much simpler and elegant formulation for study. Under a suitable non-dimensionalisation and scaling, we can even place the above equation in the final simplified form of

$$\dot{x} = \lambda - |1 - x|x,$$

where $\lambda > 0$ (for temperatures in the usual configuration, with $T_2 - T_1 > 0$, and a suitably defined function $x = x(t)$).

CHAPTER 23

Stommel's box model

In his work (Stommel, 1961) did not account for evaporation or precipitation, so there is no virtual salt flux, $H = 0$, but each box does exchange salinity with its surrounding environment, so $d > 0$. The set of four differential equations presented in Chapter 22 (Equation 22.6) can be reduced two coupled differential equations for the mean temperature and salinity. We notice that if we subtract the pairs of equations, then we can write

$$\begin{aligned}\frac{d\Delta T}{dt} &= c(\Delta T^* - \Delta T) - 2|q|\Delta T, \\ \frac{d\Delta S}{dt} &= d(\Delta S^* - \Delta S) - 2|q|\Delta S,\end{aligned}\tag{23.1}$$

where $\Delta T^* = 2T^*$, $\Delta S^* = 2S^*$, and we have defined the two new unknowns,

$$\begin{aligned}\Delta T &= T_2 - T_1, \\ \Delta S &= S_2 - S_1,\end{aligned}$$

and now, the flow q is given by

$$q = k(\alpha\Delta T - \beta\Delta S).$$

23.1 Non-dimensionalisation

We can then nondimensionalise the system by setting

$$\Delta T = \Delta T^*y, \quad \Delta S = \Delta S^*x, \quad t = [t]t',$$

where t' is nondimensional time. Annoyingly, notice that we have flipped the order of the equations, i.e. above we presented temperature and salinity in that order, and now this is associated with y and x .

For example, the equation in ΔT now becomes

$$\frac{dy}{dt'} = c[t](1 - y) - 2|q|[t]y$$

and we see that we should select

$$[t] = \frac{1}{c}.$$

If we now examine the remaining factor:

$$2|q|[t] = \frac{2k\alpha\Delta T^*}{c} \left| -y + \frac{\beta\Delta S^*}{\alpha\Delta T^*} \right| = \frac{1}{\lambda}(Rx - y),$$

we can see where the constants λ and R are defined from. Notice that there has been a negative sign flip within the absolute values.

Dropping the primes henceforth, we have the following set of non-dimensional equations to study for the unknowns, $x = x(t)$, and $y = y(t)$:

$$\begin{aligned} \frac{dx}{dt} &= \delta(1 - x) - |f(x, y)|x, \\ \frac{dy}{dt} &= 1 - y - |f(x, y)|y, \end{aligned} \tag{23.2}$$

where we have introduced the function,

$$f(x, y; R, \lambda) = \frac{1}{\lambda}(Rx - y), \tag{23.3}$$

where there are now three non-dimensional parameters given by

$$\begin{aligned} \delta &= d/c, \\ \lambda &= c/(2\alpha k\Delta T^*), \\ R &= \beta\Delta S^*/(\alpha\Delta T^*). \end{aligned}$$

Together Equation 23.2 and Equation 23.3 form a system of equations for $(x(t), y(t))$, with parameters δ , λ , and R .

Although there are many parameters, it is important to note that the quantity f is crucial since it corresponds to the direction of flow in the bottom pipe of the Ocean model. By convention, the system was set up so that $f > 0$ corresponds to flow through the bottom pipe goes in the direction of the equator (Box 1) as a result of higher densities at the high latitudes. We are subsequently interested in whether it is possible for f to switch sign, which would correspond to a bottom flow going from equator to poles, and the entire system switching direction.

23.2 Equilibrium states

Let the equilibrium states be given by (x^*, y^*) with $f(x^*, y^*) = f^*$. Then setting the right hand sides of (Equation 23.2) to zero and solving, we find

$$x^* = \frac{\delta}{\delta + |f^*|} \quad \text{and} \quad y^* = \frac{1}{1 + |f^*|}. \tag{23.4}$$

We can solve for f^* in both equations and equate the result to one another. This gives

$$\delta \frac{1 - x^*}{x^*} = \frac{1 - y^*}{y^*} = |f^*|.$$

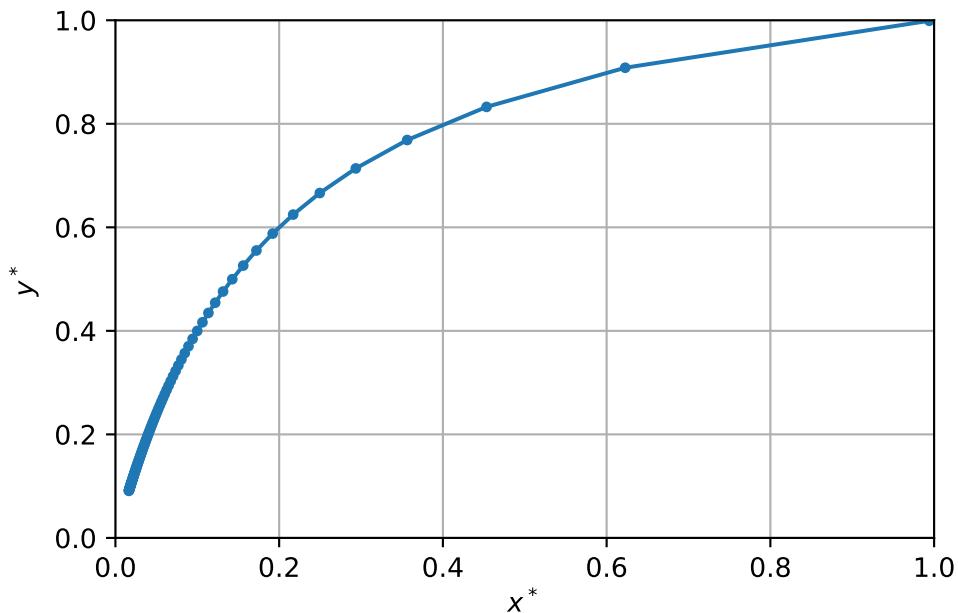
Therefore the equilibrium points lie along the above curve. Let us generate different values of f^* and plot the combination.

```

import numpy as np
import matplotlib.pyplot as plt

delt = 1/6
fmat = np.linspace(1e-3, 10, 101)
xs = delt/(delt + np.abs(fmat))
ys = 1/(1 + np.abs(fmat))
plt.plot(xs, ys, '.-')
plt.xlim((0,1))
plt.ylim((0,1))
plt.xlabel("$x^*$"); plt.ylabel("$y^*$");
plt.grid(1)

```



The above plot shows the steady-state values at $\delta = 1/6$. Each point refers to a different value of $|f^*|$. Note that as $|f^*| \rightarrow \infty$, then $(x^*, y^*) \rightarrow (0, 0)$ while as $|f^*| \rightarrow 0$, then $(x^*, y^*) \rightarrow (1, 1)$. Now the above does not tell us what equilibrium states will exist at a specific value of δ since it requires information about f^* . We take (Equation 23.3) and combine with (Equation 23.4). We conclude that the equilibrium points must satisfy

$$\lambda f^* = \phi(f^*),$$

where we have defined the function ϕ according to

$$\phi(f^*) = \frac{\delta R}{\delta + |f^*|} - \frac{1}{1 + |f^*|}. \quad (23.5)$$

Therefore, for each combination of the parameters (λ, δ, R) , we must solve $\delta f^* = \phi(f^*)$ to determine f^* . Once the values of f^* are known, then the steady-states (x^*, y^*) are also known. Here is a typical graph showing the potential intersections at the values of $\lambda = 1/5$, $\delta = 1/6$ and $R = 2$.

```

import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize as sciopt

```

```

lam = 1/5;
delt = 1/6;
R = 2;

f = np.linspace(-2.5, 2.5, 101)
phi = delt*R/(delt + np.abs(f)) - 1/(1 + np.abs(f))

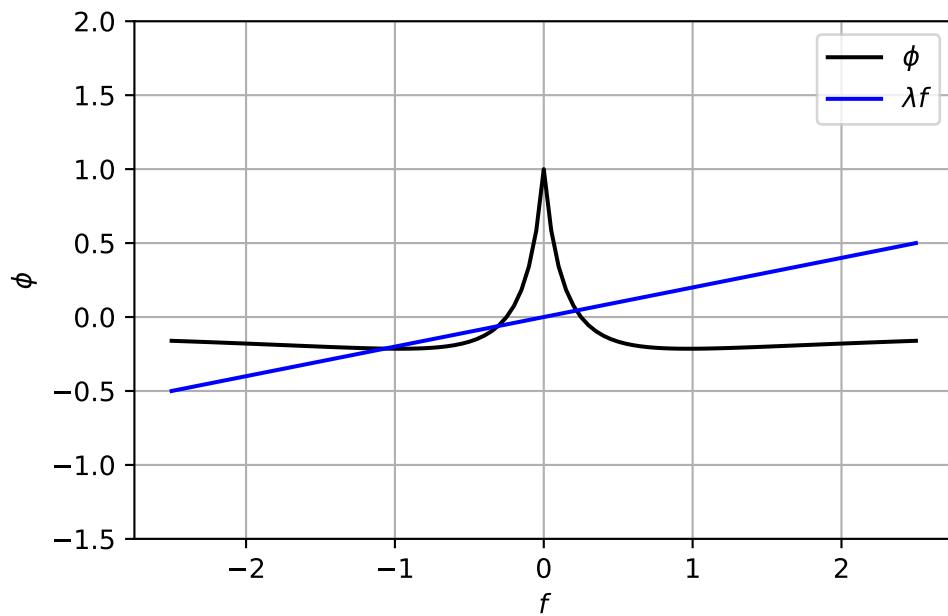
plt.plot(f, phi, 'k', label='\phi')
plt.plot(f, lam*f, 'b', label='\lambda f' )
plt.ylim((-1.5,2))
plt.xlabel("f")
plt.ylabel("\phi")
plt.legend()
plt.grid(1)

```

```

<>:12: SyntaxWarning: invalid escape sequence '\p'
<>:13: SyntaxWarning: invalid escape sequence '\l'
<>:16: SyntaxWarning: invalid escape sequence '\p'
<>:12: SyntaxWarning: invalid escape sequence '\p'
<>:13: SyntaxWarning: invalid escape sequence '\l'
<>:16: SyntaxWarning: invalid escape sequence '\p'
/var/folders/lw/7v5bfgnj6gj4qhr6fykhcp3h0000gn/T/ipykernel_88171/2976439648.py:12: Sy
    plt.plot(f, phi, 'k', label='\phi')
/var/folders/lw/7v5bfgnj6gj4qhr6fykhcp3h0000gn/T/ipykernel_88171/2976439648.py:13: Sy
    plt.plot(f, lam*f, 'b', label='\lambda f' )
/var/folders/lw/7v5bfgnj6gj4qhr6fykhcp3h0000gn/T/ipykernel_88171/2976439648.py:16: Sy
    plt.ylabel("\phi")

```



Although there are many parameters that can alter the shape of the overall graphs, note that if λ is increased from its current value of $\lambda = 1/6$, then two of the left roots will disappear, leaving only a single root.

23.3 Stability

For our purposes, we will mainly proceed using numerics, and bypass the need to study the above results analytically.

In the accompanying lecture script, `lecture29-StommelPhasePlane.ipynb`, we will obtain the eigenvalues numerically, and generate the phase plane. At the test point $R = 2$, $\delta = 1/6$, and $\lambda = 1/5$, we generate the following phase plane picture.

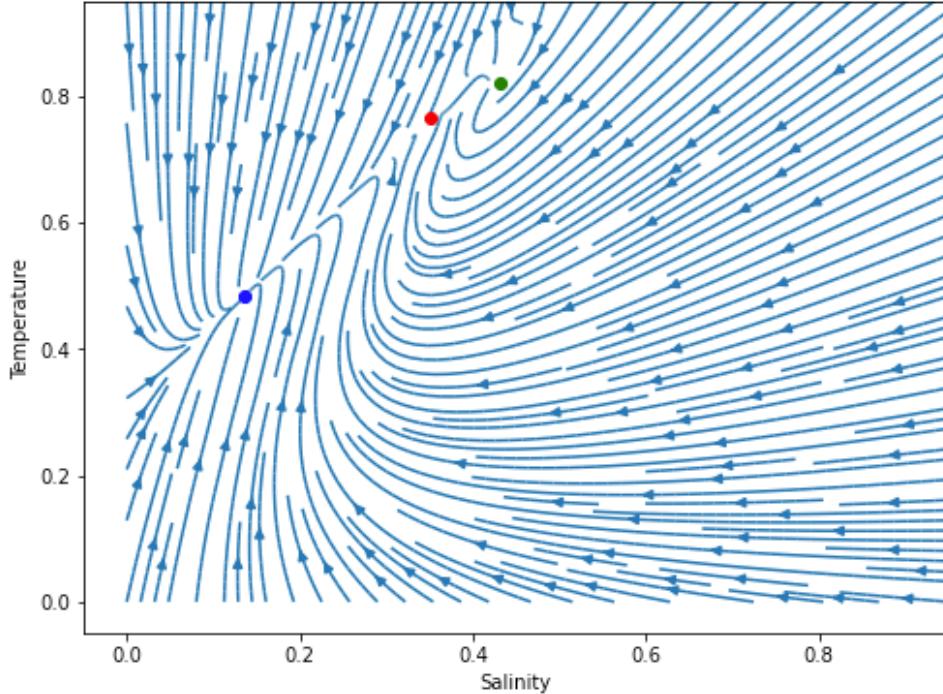


Figure 23.1: Phase plane for $R = 2$, $\lambda = 1/5$ and $\delta = 1/6$. The blue, red, and green fixed points correspond to f_1 , f_2 , and f_3 in order. The blue point is a stable node, the red point is an unstable saddle, and the green point is a stable spiral.

Details on the stability analysis

This part is only for additional detail in the course; it involves some (non-interesting) algebra so is only included here for additional context.

We let $x = x^* + \xi$ and $y = y^* + \eta$ and linearise the system about the fixed points. This gives

$$\begin{pmatrix} \dot{\xi} \\ \dot{\eta} \end{pmatrix} = A \begin{pmatrix} \xi \\ \eta \end{pmatrix},$$

where the matrix A is given by

$$A = \begin{pmatrix} -(\delta + |f^*|) \mp \frac{Rx^*}{\lambda} & \pm \frac{x^*}{\lambda} \\ \mp \frac{Ry^*}{\lambda} & -(1 + |f^*|) \pm \frac{y^*}{\lambda} \end{pmatrix}$$

if $f^* \gtrless 0$.

There are two ways of proceeding. We can numerically substitute the fixed points into the above matrix for A and then calculate the eigenvalues (numerically or otherwise). Or we can directly proceed analytically.

If proceeding analytically, we can then calculate the trace and determinant, giving

$$T = -(1 + \delta + 3|f^*|),$$

$$D = (\delta + 2|f^*|)(1 + |f^*|) \pm (1 - \delta) \frac{y^*}{\lambda}.$$

Using the above, we can analytically calculate the key discriminant expression of $T^2 - 4D$ (or numerically) in order to verify stability. The details of the classification scheme is given in the appendix Chapter 46.

23.4 Bifurcation diagrams

Once the above stability analysis, it is possible to sketch the final bifurcation diagram.

First, we can note that as λ increases, the graph in Section 23.2 indicates that there may either be three intersections (as shown) or if λ increases then two of the intersections may disappear. We can note that:

f_i^*	$\lambda \rightarrow 0^+$	$\lambda \rightarrow \infty$
f_1^*	$-\infty$	none
f_2^*	fixed point < 0	none
f_3^*	fixed point > 0	0

We can also ascertain that there should be a point, say $\lambda = \lambda^*$ where the two roots coalesce, $f_1^* = f_2^*$, and thereafter disappear. This allows us to now plot the bifurcation (λ, f^*) -plane:

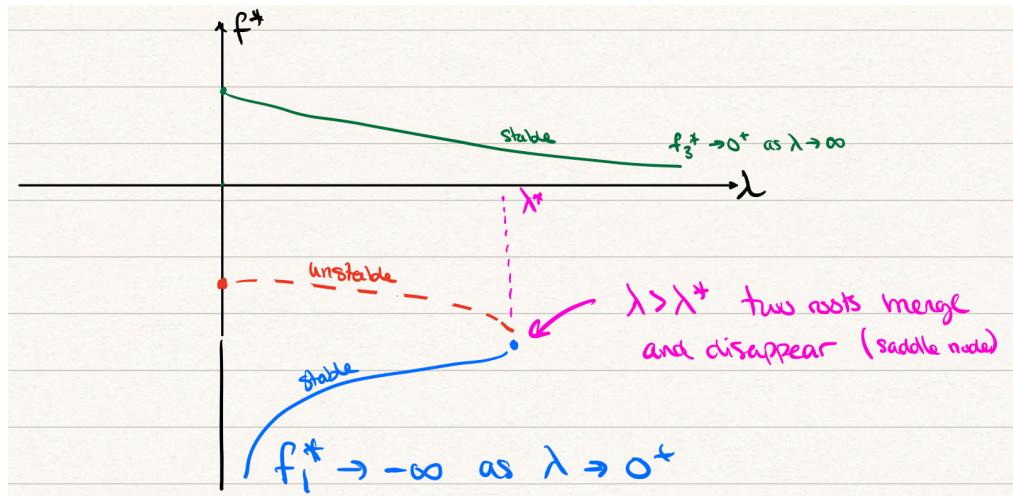


Figure 23.2

23.5 Stommel's conclusion

What is the final takeaway message(s)?

1. It is possible to pose a conceptual model of the ocean and the thermohaline circulation, which governs the exchange of hot-and-cold waters and salty waters from equatorial-to-pole zones.

2. This conceptual model suggests that tipping points and hysteresis are possible in such systems. In particular, it is possible, according to these models, for the flow directions to entirely change!

As noted by [@kaper_book]:

“Clearly, two-box models are only a caricature of the THC. At best, they account for the pole-to-equator circulation in single ocean basin (the North Atlantic). They certainly do not account for the fact that all the Earth’s oceans are connected, nor for the fact that the oceans are coupled to the atmosphere and other components of the climate system. Nevertheless, the finding that such simple models predict the possibility of two distinct stable modes of circulation, and that transitions from one more to another can be induced by changing the forcing parameters [–] has had a significant impact in oceanography and climate science.”

CHAPTER 24

Box models for flood estimation

So far, we have focused primarily (though not exclusively) on mathematical models of climate, which concerns long-term averages of weather over many years. We have focused primarily (though not exclusively) on simplified box or conceptual models. For instance, in the last few chapters, we have studied box models of the ocean, where temperatures and salinity are lumped together into different categories/regions (or boxes), with exchanges of quantities between such boxes. These can also be considered phenomenological models or process-driven models.

Similar ideas are used to model other aspects of the environment. In this chapter, we will provide an introduction on the mathematical modelling of flooding. Flooding is considered one of the most significant dangers posed around the world, and within the UK, the cost of flood damage and flood prevention is estimated at £200 billion per year.

Table 2.1 Summary of present-day flood risks and flood-management costs

	Properties at risk	Average annual damage (£ million)	Flood management costs 2003-04 (£ million)
River and coastal flooding			
England and Wales	1,740,000	1,040	439
Scotland	180,000	32 (fluvial only)	14
Northern Ireland	45,000	16 (fluvial only)	11
Intra-urban flooding			
All UK	80,000	270	320
Total	2,045,000	1,400	800

(Totals are rounded)

Figure 24.1: Table of flood damage; UK government report given in reference above

24.1 Terminology

First, examine a typical geometry in the image below.

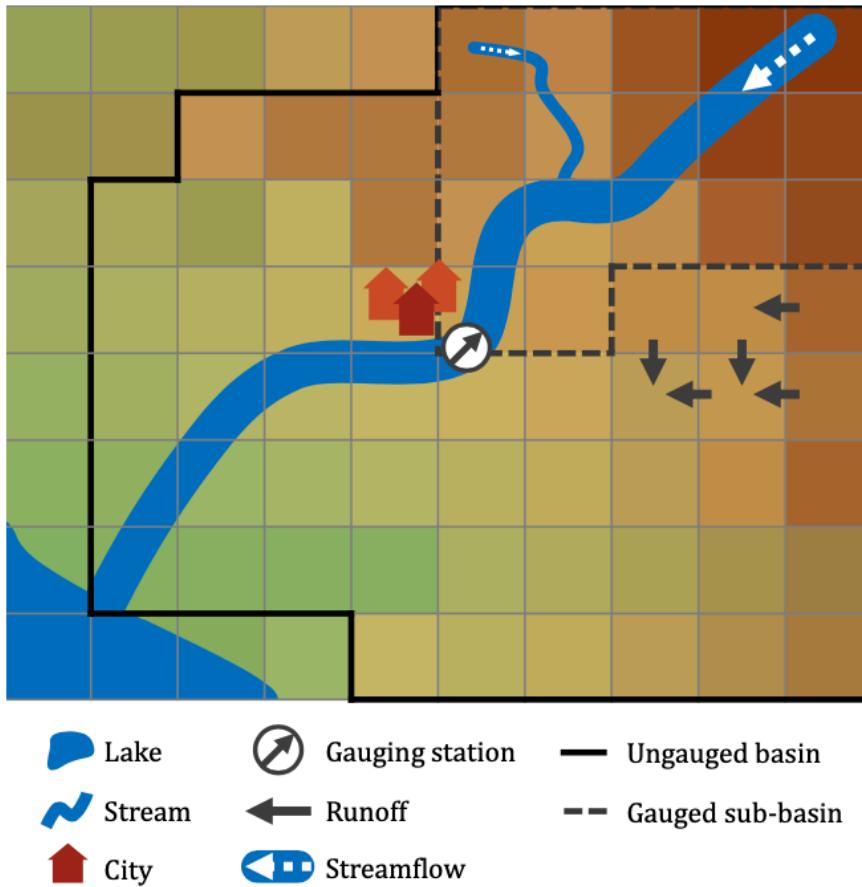


Figure 24.2: Typical geometry of consideration; image from Gauch and Lin (2020)

In flooding, hydrologists are primarily interested in two related concepts of *streamflow* and *runoff*. Streamflow is typically measured at a prominent outlet, such as the mouth of a river or the outflow to a lake, and consists of the amount of water that flows through a cross section, expressed as cubic metres per second. At any given point in the above diagram, we can also measure the runoff, which is expressed in millimetres per day (mm/d). Given the river path in the figure above, we could theoretically convert from runoff to streamflow: we for each grid, we multiply the runoff by the surface area, which yields the volumetric per time quantity of streamflow.

In the above diagram, the upstream area of an outlet (the lake or the gauging station, for instance), is called a *basin*, *catchment*, or *watershed*.

For reference, these models are re-interpreted and cleaned up versions of what appears in the work by [Jothityangkoon et al. \(2001\)](#).

24.2 A basic flood model

The basic flood model or rainfall-runoff model is essentially a water balance equation. Here is a simple model with one store and one parameter. We consider the total water exchange that occurs in a given area. Precipitation input, $P(t)$, measured in mm/day is obtained. The quantity $S(t)$, measured in mm, then measures the current “storage” in the soil moisture. Actual evaporation $E_a(t)$ will remove moisture from the soil. Then:

$$\frac{dS}{dt} = P(T) - Q_{se}(t) - E_a(t). \quad (24.1)$$

The makeup of the right hand-side are as follows.

The term Q_{se} measures the saturation excess overland flow. This is essentially the quantity that arises when the soil cannot support additional moisture, and produces runoff. We can consider it modelled as:

$$Q_{se}(T) = P(t)\sigma(S - S_{\max}),$$

where σ is the sigmoid or activation function. For instance, we might take

$$\sigma(S - S_{\max}) = H(S - S_{\max}) = \begin{cases} 1 & S > S_{\max} \\ 0 & S < S_{\max} \\ 1/2 & S = S_{\max}, \end{cases}$$

where H is the Heaviside step function, which for convenience takes the midpoint value when evaluated at zero.

The above model for Q_{se} makes some sense. If the soil has not exceed some ‘maximum soil moisture storage’ value S_{\max} , then there is no saturation excess and the total runoff is zero. However, once this threshold is exceed, the precipitation is routed directly to the runoff.

The actual evaporation (or more accurately evapotranspiration) includes interception loss from plant canopies at the land surface, and combined evaporation and transpiration taken out of the soil water. We take it to be written in a fashion proportional to the total soil moisture:

$$E_a(t) = \frac{E_p(t)}{S_{\max}} S(t).$$

We assume that the metereological data provides estimates of the potential evaporation, . You do not have to know about the specifics of the difference between actual and potential evaporation, though see a description of how these two quantities are measured at [this link](#). Above, it is nice to think of the fact that the dimensions of must be measured in storage units (mm) per time. The quantity

$$\frac{S_{\max}}{E_p},$$

is the approximate time it would take for a bucket with S_{\max} to completely evaporate.

Therefore, on the assumption that metereological data provides values of precipitation P and potential evapotranspiration (or simply evaporation), we can use the above to calculate the soil moisture S . The key quantity to predict is the runoff, which in this case is simply the saturation excess:

$$\text{runoff} = Q_{se}(t).$$

In summary, the above model consists of a single store and a single parameter (S_{\max}). In theory, you would obtain time-series data of streamflow or runoff, and then attempt to calibrate the parameter.

24.3 A flood model with slow and fast flows

If the above model is tested on real data, we would typically observe that it performs well in predicting long-time runoffs, but that it does not cope very well with inter-annual variations. Indeed, it fails to capture a key mechanism of water transfer, which suggests that

runoff should be separated into two components: a delayed subsurface runoff, $Q_{ss}(t)$, when soil moisture storage in the bucket exceeds some field capacity threshold (think of this as groundwater flow); and the previous saturation excess (think of this as overland flow).

In addition, we might want to separate the evapotranspiration as

$$E_a = E_b + E_v,$$

where E_b measures the bare soil evaporation while E_v measures the transpiration due to vegetation. In theory, this would allow us to partition land into soil-covered or vegetation-covered.

The model is then

$$\frac{dS}{dt} = P(t) - [Q_{se}(t) + Q_{ss}(t)] - [E_b(t) + E_v(t)].$$

First, the saturation excess overland flow is as it was previously:

$$Q_{se}(t) = P(t)\sigma(S - S_{\max}),$$

where again for simplicity, we could take $\sigma = H$ and use the Heaviside step function as the activation function..

We also allow now for the subsurface flow, $Q_{ss}(t)$, (again in mm/day), modelled via

$$Q_{ss}(t) = a(S - S_{fc})\sigma(S - S_{fc}),$$

which produces flow proportional to how much the soil has exceeded its field capacity value, S_{fc} , subject to a proportionality constant a .

As for the two evaporation rates, they are subsequently weighted by a factor dependent on M , the forest fraction parameter, with $0 < M < 1$. We do not go into detail on their forms, but refer the reader to the original work by [Jothityangkoon et al. \(2001\)](#), from which this model was taken. The evaporation rates are,

$$E_b = (1 - M)E_p(t)\frac{S}{S_{\max}}.$$

The above is the usual expression for the evaporation rate, but additional weighted by the $(1 - M)$ factor, which is zero if the area is entirely forest-covered. Next, the vegetation-covered evaporation is,

$$E_v = \begin{cases} ME_p & S > S_{fc}, \\ \frac{S}{S_{fc}}ME_p & S < S_{fc}, \end{cases}$$

and so the model depends on whether the groundwater region is entirely saturated or not. If the groundwater region is entirely saturated, then the maximal transpiration occurs, specifically with $E_v = E_p$ for the completely forest-covered region. On the other hand, if the groundwater region is not completely saturated, then we allow the vegetative transpiration rate to be a fraction of E_p , dependent on the soil moisture storage S .

Again, the runoff is calculated via

$$Q = Q_{se} + Q_{ss}.$$

This model has one store and 4 parameters: S_{\max} (mm), S_{fc} (mm), a (1/day), and $0 < M < 1$. Like all other models, you can attempt to train the model by processing known data and fitting the parameters.

24.4 Calibration and analysis

Non-examinable

The following is presented, but is not examinable. Still, it is interesting to know briefly how the above hydrological models can be used in practice.

The procedure, which is implemented via code provided in the accompany paper by [Trotter et al.](#) is now as follows. We obtain calibration data, primarily of precipitation $P(t)$, potential evapotranspiration $E_p(t)$, and streamflow or runoff.

As an example, we use 5 years worth of climate and streamflow data from Buffalo River near Flat Woods, Tennessee, USA, to illustrate. The catchment was randomly selected from those provided within the CAMELS dataset [Addor et al. \(2017\)](#). The USGS gauge ID for this catchment is 3604000.

When testing models' performance against real data, hydrologists often use a range of metrics to evaluate and compare these models. One of the most commonly used is the [Nash-Sutcliffe efficiency \(NSE\)](#). NSE acts like a mean squared error (MSE) between observed and simulated flow discharge. It is calculated by taking one minus the ratio between the error variance of the model time series and the error variance of the observed time series of flow discharge. We have

$$\text{NSE} = 1 - \frac{\sum_{t=1}^T (Q_o^t - Q_m^t)^2}{\sum_{t=1}^T (Q_o^t - \bar{Q}_o)^2}$$

where \bar{Q}_o is the mean of observed discharges for T data points and Q_m^t , Q_o^t are the model discharge and observed discharge at time t . Note that $\text{NSE} \in (\infty, 1]$, with values closer to 1 indicating the model has a great predictive skill for flow discharges based on the observed data. When $\text{NSE} = 0$, the model has the same predictive skill as the mean of the time series in terms of the sum of the MSE. For $\text{NSE} < 0$, the observed mean is a better predictor for the model and indicates the model is not performing well.

We won't discuss how this is done in detail, but the essence is that we apply the flooding models above with approximated parameter values and obtain the simulated data Q_m^t . The ODEs can be integrated using a normal integrator (similar to Euler's method or perhaps something more sophisticated). Then during a predetermined 'calibration' period (in the below, roughly in the period 1990-1991), we apply an algorithm to determine the 'best fitting' parameters of the above model, whose values will hopefully minimise the objective function (given by the above NSE). Once this calibration is done, and the parameters determined, one can evaluate the NSE over a different 'evaluation' period to determine how well the model fits. The specific algorithm used below is called the covariance matrix adaption evolution strategy (CMA-ES, [reference](#)). CMA-ES is a stochastic, derivative-free, numerical optimisation algorithm.

The basic model

The basic model with the dataset converges to the parameters of $S_{\max} \approx 0.69$.

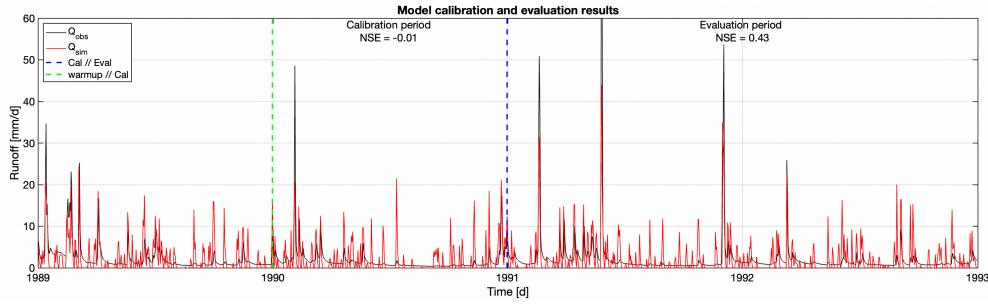


Figure 24.3: Calibration and testing for model 1

And here is a blowup over a one-year period after the calibration.

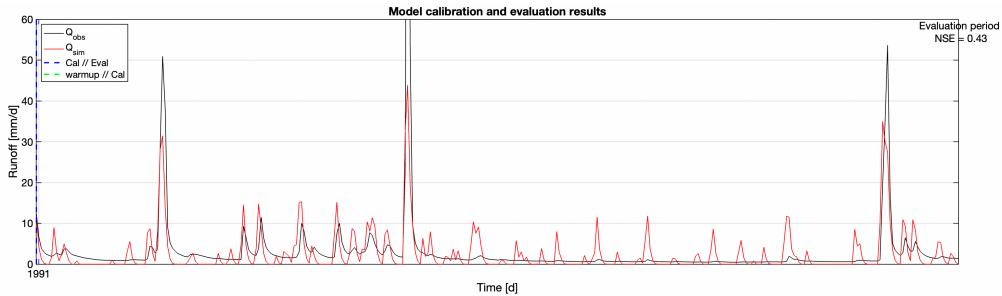


Figure 24.4: Zoom of the above

The slow-fast model

The more complicated model converges to the parameters of $S_{\max} \approx 963$, $S_{fc} \approx 0.78$, $a \approx 0.03$, and $M \approx 0.95$.

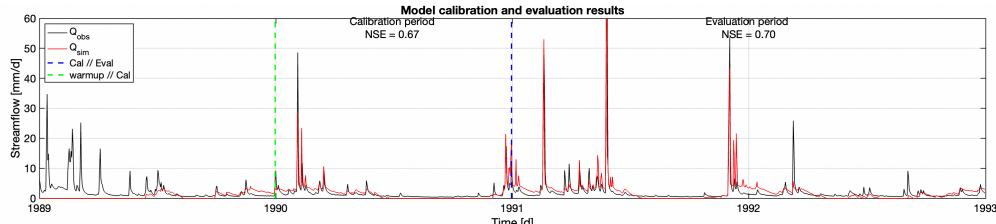


Figure 24.5: Calibration and testing for model 2

And here is a blowup over a one-year period after the calibration.

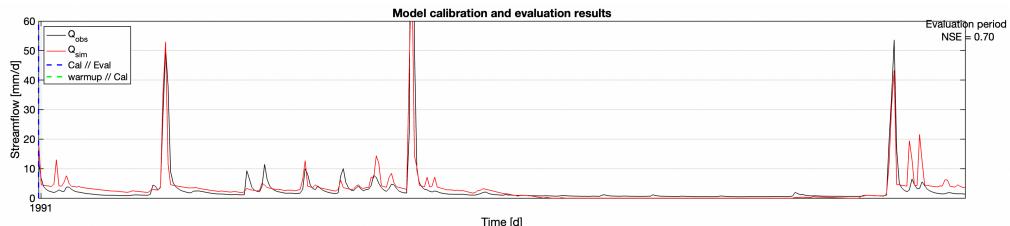


Figure 24.6: Zoom of the above

Part VI

Exercises

CHAPTER 25

Problem set 1

The intention of this problem set is to practice concepts from material related to conservation laws and non-dimensionalisation. Although these concepts seem quite separate from “Maths of Planet Earth”, actually, they form important pillars of mathematical modelling.

Q1. Bump lemma

Prove the following one-dimensional lemma, which was used in the derivation of the heat equation.

If $\int_a^b g(x) dx = 0$ for all a and $b \in [0, 1]$, then $g(x) \equiv 0$ for all $x \in [0, 1]$.

Hint: think of a proof by contradiction.

Q2. A source in the heat equation

Consider the same heat experiment discussed in Chapter 2 but now consider a bar that has an internal source or sink generating or removing heat, such as the case of a boiler with an internal heating element. By adapting a similar derivation to the one presented, explain why the modified conservation of heat equation is

$$\frac{\nabla}{\nabla t} \int_a^b \rho c T dx = q(x = a, t) - q(x = b, t) + \int_a^b R(x, t) dx.$$

In addition:

- a. By studying the dimensions of the other terms in the above equation, find what the dimensions of R are. What does $R > 0$ mean and $R < 0$?
- b. Hence derive the partial differential equation that governs the temperature T .
- c. By introducing the appropriate scalings on each of the variables, x , t , and T , non-dimensionalise the PDE and discuss the non-dimensional parameters (there will be two).

Q3. The unique timescale in the heat equation

During our investigation of the heat equation, we found that it was possible to scale time so as to scale out the only non-dimensional parameter that appears in the PDE (II). This produced (Equation 3.1). The disappearance of all non-dimensional parameters is due to the fact that only a single sensible timescale exists.

By adjusting the boundary conditions, we may create a new problem involving heat flow where a unique ‘special’ timescale can no longer be chosen.

Consider a system where one side of the rod is heated in some periodic fashion, e.g. set the initial and boundary conditions to be

$$\begin{aligned} T(x, 0) &= T_0 \\ T(0, t) &= T_a \cos(\omega t), \\ T(L, t) &= T_b. \end{aligned}$$

- a. What must the units of ω be?
- b. Non-dimensionalise as usual and, without selecting the timescale, $[t]$, identify the key non-dimensional parameters that remain. Write a brief sentence to describe their physical interpretation.
- c. There are two sensible choices for setting the timescale, $[t]$. Identify the two choices and present the reduced set of equations in each case.

Q4. Timescale in the surface energy

Take the basic zero-dimensional energy model studied in (Equation 4.7) for the temperature of the troposphere:

$$C \frac{dT}{dt} = \frac{1}{4}Q(1-a) - \sigma\gamma T^4.$$

- a. Non-dimensionalise the model by choosing $T = [T]T'$, $t = [t]t'$, and $Q = [Q]Q'$. Show that it is possible to select the scalings on the temperature and time so as to completely remove all constants from the problem when Q is assumed to be constant.
- b. Thus, show that the analysis of the above equation is equivalent to studying

$$\frac{dT}{dt} = 1 - T^4,$$

where we have dropped the primes and assumed that the albedo is such that $1-a \neq 0$.

- c. From your choice of $[t]$, estimate the typical dimensional value using $d \approx 10\text{km}$, $\rho \approx 1\text{kg m}^{-3}$, $c_p \approx 10^3\text{J kg}^{-1}\text{K}^{-1}$.

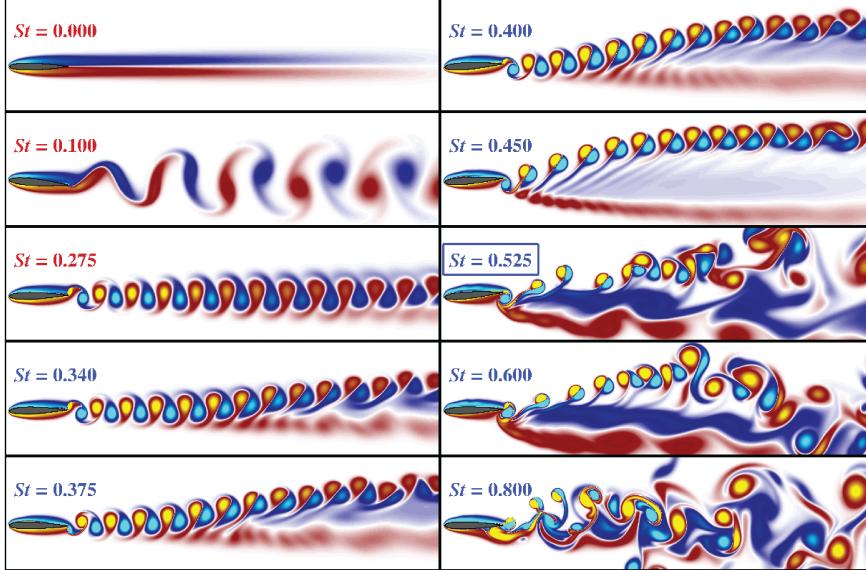
Use a pocket calculator to verify your calculations and conclude that this timescale is on the order of a month. What is the relevance of this approximation as it concerns the steady-state solution?

Q5. Choice of scalings in Navier Stokes

Suppose that the time, with $t = [t]t'$ is non-dimensionalised as $[t] = \frac{1}{f}$ where f is a frequency. Show that the Navier-Stokes equations, introduced in Problem Class 2, can be written as

$$\frac{\partial u}{\partial t} + \frac{1}{St} (u \cdot \nabla) u = -\nabla p + \frac{1}{Re} \nabla^2 u.$$

Above, the Reynolds number is defined via $Re = \frac{fL^2}{\nu}$ and the Strouhal number is defined via $St = \frac{Lf}{U}$. The Strouhal number represents the effect of vortex shedding in flow across an object. The higher St is, the more vortices will be generated by the flow.



By considering the SI units for the different quantities of f , L , ν , and U , confirm that the two non-dimensional numbers introduced above are indeed non-dimensional.

CHAPTER 26

Problem set 2

The intention of this problem set is to practice concepts from Chapter 5 (asymptotic approximations of algebraic equations) to Chapter 7 (Euler's method and numerical solutions of differential equations). These techniques form some of the most powerful techniques at your disposal in applied maths.

26.1 Getting started with Noteable

Use your [Moodle course page](#) to access the Noteable Python interface by clicking on the appropriate link in the course materials. Follow the instructions on the Moodle page to add the Git repository to your directory.

1. Navigate to the welcome screen for MA30287 and follow the instructions in `Welcome.ipynb` and `FirstTimeSetup.ipynb` in order to create your own local directory at `/MA30287_workspace/`
2. Navigate to the workspace folder. Click `New -> Notebook`. If asked, select the kernel `Python 3 (ipykernel)`.
3. In the first line of input, select, in toolbar `Code -> Markdown`. This allows you to annotate your notebook with Markdown-style text input.

In the first line of input, type `# Problem set 2`. Then either type `Shift + Enter` or press the play button which will execute the line(s) of input. Your markdown text should render as a nicely formatted entry.

4. Rename the file to something appropriate, like `ps02_scripts`. You can do this by right-clicking the filename in the file manager and selecting `Rename`.

Proceed to the next question.

26.2 Testing the solutions of a cubic

In this question, you will develop the numerical solutions for the roots of the cubic equation:

$$\epsilon x^3 - x + 1 = 0, \quad \epsilon > 0$$

Type the following code into your Jupyter notebook. This code uses a command, which you will learn in a later week, called `fsolve`, in order to solve nonlinear equations.

When writing the below code, you may want to separate the import commands into their own input field in the notebook.

```
import numpy as np
from scipy.optimize import fsolve

ep = 0.1          # epsilon value
xguess = 1.1      # Initial guess of root

f = lambda x: ep*x**3 - x + 1
xsol = fsolve(f, xguess)
print("Solved root at x = ", xsol)
```

Solved root at x = [1.15346731]

26.3 Analysis of singular cubic equation

Consider the cubic equation

$$\epsilon x^3 - x + 1 = 0,$$

with $\epsilon \ll 1$ and $\epsilon > 0$.

1. Develop the first three terms of an asymptotic expansion about the root by setting

$$x = x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots$$

2. Fill out the following table.

ϵ	x_{exact}	$x_{\text{exact}} - x_0$
0.1	—	—
0.08	—	—
0.06	—	—
0.04	—	—
0.02	—	—

Use your code in Section 26.2 to input numerical approximations to the above entries.

Create a graph by hand of the data (it does not have to be extremely accurate), as plotted in the $(\epsilon, x_{\text{exact}} - x_0)$ -plane. Fit a line to this graph and estimate the gradient. Is this consistent with what you derived above?

3. By rescaling x appropriately in terms of ϵ , derive the first three terms of the asymptotic approximations of the remaining roots.

26.4 A damped projectile problem

In Chapter 6 you performed the asymptotic analysis for a projectile. The small parameter was ϵ and represented $v_0^2/(gR_E)$ (a parameter that includes the initial velocity, v_0 , gravity g , and the radius of the Earth, R_E).

If air resistance is included, then the non-dimensional toy model is instead

$$\begin{aligned}\frac{d^2y}{dt^2} &= -\frac{1}{(1+\epsilon y)^2} - \frac{\alpha}{(1+\epsilon y)} \frac{dy}{dt}, \\ y(0) &= 0, \\ y'(0) &= 1.\end{aligned}\tag{26.1}$$

where $\alpha \geq 0$ is the parameter that controls air resistance.

1. Begin by assuming that α is a fixed number and consider the limit where $\epsilon \ll 1$. Find a **one-term asymptotic expansion** of the solution for small ϵ .
2. (**Challenging**) Is the effect of the air resistance to increase or decrease the flight time? Justify based on your analytical solution.

26.5 ODE solvers and Euler's method

Return to the setup of the above question.

1. Modify the script shown in Section 6.2 in order to solve the initial-value problem from the previous question at a prescribed value of ϵ and α .
2. Using a **pocket calculator** (or your phone calculator) apply Euler's method with $\Delta t = 0.2$, $\epsilon = 0.2$, and $\alpha = 1$ to determine the position of the projectile at $t = 0.6$.
3. Compare your hand calculation with the result from the Python output, as well as with your asymptotic approximations.

CHAPTER 27

Problem set 3

In Chapter 8 we discussed how the zero-dimensional energy equation occurs in the form of

$$(\rho c_p V) \frac{dT}{dt} = E_{\text{in}}(t, T) - E_{\text{out}}(t, T).$$

This is oversimplistic since in actuality, your temperature function, T , should also depend on space. For instance, if you are interested in modelling T with a depth coordinate, then it would be $T(z, t)$. In this case, we know that temperature would be governed by a second-order differential equation (in space) such as the kind that appears in (Equation 2.1).

In this problem set, we will make up a toy example of a differential equation that might be encountered in heat flow. Consider now (in non-dimensional coordinates):

$$\frac{\partial T}{\partial t} = \epsilon \frac{\partial^2 T}{\partial z^2} + 2 \frac{\partial T}{\partial z} + T = 0,$$

which is determined for some function $T(z, t)$. Here, the small parameter $\epsilon > 0$, $\epsilon \ll 1$, is linked to the heat diffusion.

We will study the steady-state version of the problem. So consider only $T = T(z)$. We will make up some boundary conditions as well.

$$\begin{aligned} \epsilon T'' + 2T' + T &= 0, \\ T(0) &= 0, \\ T(1) &= 1. \end{aligned} \tag{27.1}$$

The challenge is to study the above problem for small values of ϵ .

Q1. Conversion to first-order system

By using the procedure reviewed in Section 6.2, convert (Equation 27.1) to a first-order system of equations.

Q2. Numerical solutions

By adapting the code studied in lectures (script `lecture12-SolvingBVPs`), write a numerical code to solve (Equation 27.1) using Python's built-in functions. Use your code to investigate the solution profiles for different values of ϵ .

Q3. Investigation of the boundary layer

Using your code, use the following command in order to investigate the maxima (x_m, T_m) as ϵ varies:

```
ind = np.argmax(Y[0])
print(ep, z[ind], Y[0, ind])
```

You may want to consider, as an example, the values $\epsilon = \{0.05, 0.1, 0.15, 0.2\}$ and fill the following table.

ϵ	x_m	T_m
0.05		
0.1		
0.15		
0.2		

Create a plot of (ϵ, x_m) and discuss the observed trend and its implications.

Q4. Outer asymptotic solutions

Begin by setting

$$T = T_{\text{outer}} = T_0(z) + \epsilon T_1(z) + \epsilon^2 T_2(z) + \dots$$

Substitute the above expansion into the system and solve for the first two orders.

You may verify that the solution is given by

$$T_0 = e^{1/2} e^{-z/2}, \quad (27.2)$$

$$T_1 = -\frac{1}{8} e^{1/2} e^{-z/2} (z - 1). \quad (27.3)$$

Q5. Inner asymptotic solutions

There will be a boundary layer near $z = 0$. Set $z = \epsilon^\alpha s$ and $T(z) = U(s)$. Follow the same procedure, as in Chapter 8 in order to determine the correct choice of α for the inner region. This choice should balance the two terms $\epsilon T''$ and $2T'$.

Q6. Matching and comparison

Expanding the inner solution as $U = U_0(s) + \epsilon U_1(s) + \dots$, write down the equation and boundary conditions that U_0 must satisfy. You will notice that U_0 is governed by a second-order differential equation and therefore needs two boundary conditions. One boundary condition comes from $z = 0$, i.e.

$$U_0(0) = 0.$$

The other boundary condition is a matching condition:

$$\lim_{s \rightarrow \infty} U_0(s) = \lim_{z \rightarrow 0} T_0(z),$$

which imposes that the inner solution, as it leaves the boundary layer, matches the outer solution, it tends into the inner region.

Solve for U_0 .

CHAPTER 28

Problem set 4

Note

Note that Q1 and Q4 are new to complement the coverage on numerical solutions of PDEs.⁴

Q1. Finite difference formulae

The finite-difference formulae used in our numerical ODE and PDE schemes can be derived using Taylor series approximations. Consider a function U that can be expanded as a Taylor series.

- Show that

$$U(x+h) = U(x) + hU'(x) + \frac{h^2}{2}U''(x) + \frac{h^3}{6}U'''(x) + \dots,$$
$$U(x-h) = U(x) - hU'(x) + \frac{h^2}{2}U''(x) - \frac{h^3}{6}U'''(x) + \dots.$$

- By subtracting the two equations above, derive the centred difference formula used for the first-derivatives in our algorithms:

$$U'(x) = \frac{U(x+h) - U(x-h)}{2h} + O(h^2).$$

This is what's known as the centred difference formula for the derivative.

- By adding the two Taylor series equations above, derive the centred difference formula used for the second-derivatives in our algorithms:

$$U''(x) = \frac{U(x+h) - 2U(x) + U(x-h)}{h^2} + O(h^2).$$

- The centered difference formula for $U'(x)$ will not immediately work if it is applied to find the derivative at the first mesh point in a discretisation, e.g. $x = x_0$. Show that the forward difference formula at $x = x_0$ can be derived as

$$U'(x_0) = \frac{-3U(x_0) + 4U(x_1) - U(x_2)}{2h} + O(h^2).$$

Hint: consider as well the Taylor series for $U(x+2h)$ and then combine the expansions for $U(x+h)$ and $U(x+2h)$ to derive the result.

Q2. The wine cellar problem I

From Chapter 12, we have the heat system:

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial x^2},$$

$$T(x = 0, t) = A \cos(\omega t),$$

$$|T(x, t)| \text{ is bounded as } x \rightarrow \infty.$$

for $x \geq 0$. The initial condition can be ignored.

The solution was sought in the form of $T(x, t) = G(t)H(x)$ using a separation of variables procedure leading to $\frac{G'}{G} = \kappa \frac{H''}{H} = \lambda$. Go through an argument that the cases of $\lambda = 0$ and λ real lead to incompatible solutions. Review the calculation that $\lambda = i\omega$ leads to the desired solution given in the lecture notes.

Q3. The wine cellar problem II

Review the code written on the Noteable repository that implements the explicit Euler scheme for the wine cellar problem.

- Verify how well the numerical solution matches the exact analytical solution by overlaying the numerical solution with the analytical solution.
- Investigate the solutions under different choices of parameters and time-stepping tolerances until you are confident you know how the numerical algorithm works.

Q4. von Neumann analysis

Use von Neumann analysis to determine the stability of the explicit Euler scheme for the 4th order diffusion equation:

$$u_t = -u_{xxxx}.$$

You may use the fact that the centred difference approximation of the fourth derivative is:

$$u_{xxxx}(x_j) \approx \frac{u_{j-2} - 4u_{j-1} + 6u_j - 4u_{j+1} + u_{j+2}}{\Delta x^4}.$$

The stability criterion should be

$$\Delta t \leq \frac{\Delta x^4}{8}.$$

Q5. Variable Sun output

Satellite data indicates that Q , varies roughly between 341.37 W/m² and 341.75 W/m², with a period of about 11 years.

- Use the simple EBM (Equation 13.3), given by

$$Q(1 - a) = \sigma \gamma T^4,$$

with a constant albedo, $a = 0.3$ and greenhouse gas factor $\gamma = 0.6$ to estimate the resultant variation (max and min) in the Earth's mean surface temperature T .

- b. Similar to (a) but this time, use the Budyko balance equation,

$$Q(1 - a) = A + BT$$

with $A = 203.3 \text{ W m}^{-2}$ and $B = 2.09 \text{ W}/(\text{m}^2 \text{ }^\circ\text{C})$ to estimate the resultant variation in the surface temperature. Use $a = 0.3$.

- c. The actual variation in surface temperature is in fact less than what you computed above. Why might this be?

Q6. Phase line analysis

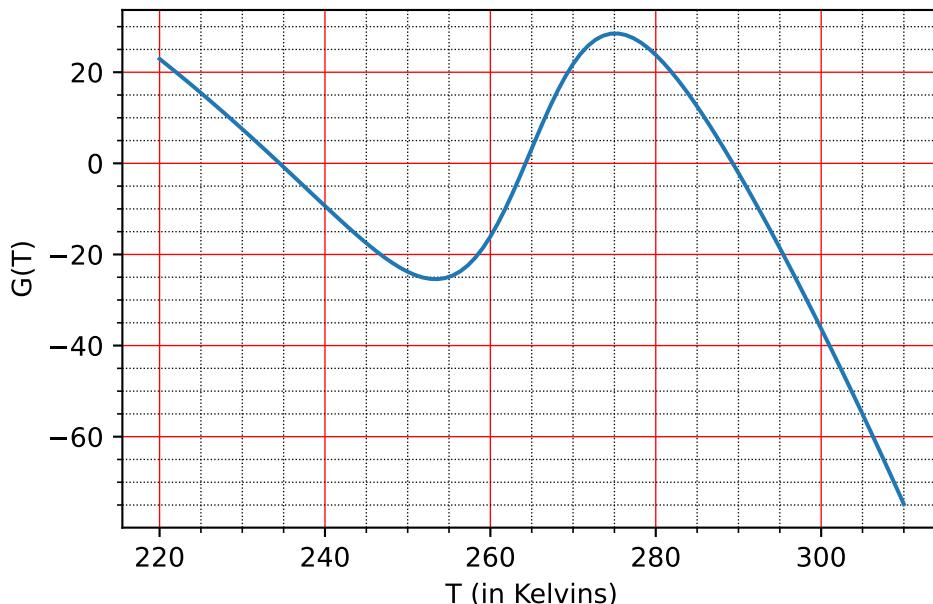
Consider the energy balance equation

$$C \frac{dT}{dt} = Q(1 - a(T)) - \sigma\gamma T^4 \equiv G(T). \quad (28.1)$$

with a given by (Equation 13.2). Because the differential equation is autonomous, we can apply phase-line analysis in order to qualitatively understand the evolution. Below is a plot of the function G :

```
import numpy as np
import matplotlib.pyplot as plt

Q = 342; sigma = 5.67e-8; gam = 0.6;
a = lambda T: 0.5 - 0.2*np.tanh(0.1*(T-265));
T = np.linspace(220, 310, 100);
G = Q*(1-a(T)) - sigma*gam*T**4
fig, ax = plt.subplots()
ax.plot(T, G)
ax.grid(); ax.minorticks_on();
# Customize the major grid
ax.grid(which='major', linestyle='-', linewidth='0.5', color='red')
# Customize the minor grid
ax.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.xlabel('T (in Kelvins)'); plt.ylabel('G(T)');
```



- a. Sketch the solution $T(t)$ of this equation for $t > 0$ if $T(0) = 230, 240, 260, 270$ and 300 .
- b. Sketch the solution $T(t)$ of this equation for $t > 0$ if $T(0) = 285$. Then sketch the solution of this equation with the same initial data in the same coordinate system if C is twice as large. Explain your answer.
- c. If γ is decreased due to the increased greenhouse effect, the entire curve is shifted upwards. Sketch the solution if $T(0) = 280$. Sketch the solution with the same initial data if γ is decreased. Explain your answer.

CHAPTER 29

Problem set 5

Q1. Evolution

Consider again the basic time-dependent EBM given in (Equation 28.1). Let T^* be a steady-state solution and set $T = T^* + u(t)$ where $u(t)$ is a small perturbation from the steady state.

- a. Show that the perturbation satisfies

$$Cu = -Du + O(u^2).$$

and hence solve for the general solution of the leading-order perturbation (ignoring quadratic terms). What are the conditions on T^* so that the steady state is linearly stable?

- b. Assuming T^* is linearly stable, find the typical response time to a perturbation. For instance, what is the time it takes for the perturbation to reach the value $u(t) = 0$ if $u(0) = 1$? How does this response time change with C ? What is the physical interpretation of this regarding the climate?

Q2. Integral of energy over the planet

Ignoring the effects of albedo, the total radiation absorbed over the surface of the planet (per unit time) is given by

$$\iint_{\text{planet}} Qs(y = \sin \varphi) dS.$$

This is what is known as a surface integral (Section 47.1). In the case of the spherical coordinate system, this is calculated by

$$\int_{\theta=0}^{2\pi} \int_{\varphi=-\pi/2}^{\pi/2} Qs(y = \sin \varphi) R_E^2 \cos \varphi d\varphi d\theta.$$

Use the properties of $s(y)$ in (Equation 14.3) to conclude that the total radiation absorbed is $4\pi R_E^2 Q$.

Q3. Mean temperature in the latitude-dependent EBM

Consider now the latitude-dependent EBM

$$C \frac{\Delta T}{\Delta t} = Qs(y)[1 - a(y)] - (A + BT) + k(\bar{T} - T).$$

Recall the albedo is given by $a = a_i$ for $y > y_s$ and $a = a_w$ for $y < y_s$.

- a. By integrating the above equation over $y \in [0, 1]$, show that the mean temperature is given by

$$C \frac{d\bar{T}}{dt} = Q(1 - \bar{a}) - (A + B\bar{T}), \quad (29.1)$$

where

$$\bar{a} = \int_0^1 s(y)a(y) dy = \alpha_w \int_0^{y_s} s(y) dy + a_i \int_{y_s}^1 s(y) dy.$$

- b. In the case that s is given by (Equation 14.3), show that

$$\bar{a} = a_i + (a_w - a_i)y_s[1 - 0.241(y_s^2 - 1)]. \quad (29.2)$$

What is \bar{a} in the two situations of a completely ice-covered world and an ice-free world?

Q4. Sensitivity of the climate

Consider the equation for the global average temperature given in (Equation 36.1):

$$C \frac{d\bar{T}}{dt} = Q(1 - \bar{a}) - (A + B\bar{T}).$$

We would like to understand how the climate behaves under a small perturbation in the solar forcing. In general, though, the ‘constants’, A , B , and \bar{a} will depend on T . For example, their values may have been calibrated under a static situation. Therefore, let $A = A(T)$, $B = B(T)$, and $\bar{a} = \bar{a}(T)$.

Below, we drop all bars for convenience.

- a. Consider a perturbation of the solar radiation, say $Q = Q_0 + \delta$ where δ is small in comparison to Q_0 . Expand now the temperature into a series:

$$T = T_0 + \delta T_1 + \dots$$

Show that at $O(\delta)$, the perturbation is governed by

$$C \frac{dT_1}{dt} = (1 - a(T_0)) - B(T_0)T_1 - A'(T_0)T_1 - T_0 B'(T_0)T_1 - Q_0 a'(T_0)T_1.$$

- b. Consequently, show that the temperature perturbation can be written as

$$B(T_0)\tau \frac{\Delta T_1}{\Delta t} = [1 - a(T_0)] - \frac{B(T_0)}{g}T_1, \quad (29.3)$$

where

$$\begin{aligned}\tau &= \frac{C}{B(T_0)} \\ g &= \frac{1}{1-f} \\ f &= f_1 + f_2 \\ f_1 &= \frac{1}{B(T_0)} (-T_0 B'(T_0) - A'(T_0)), \\ f_2 &= \frac{1}{B(T_0)} (-Q_0 a'(T_0))\end{aligned}$$

The parameter τ measures the time scale of the climate system's thermal inertia. It involves the thermal inertia of the atmosphere and also the much larger inertia of the oceans. The factor g is the climate gain; it amplifies any response to the radiative perturbation by a factor of g . Also f_1 incorporates the effect of water-vapor feedback and f_2 that of ice and snow albedo feedback.

- c. Consider (Equation 36.5) at steady state, so therefore the perturbed equilibrium temperature is equal to

$$\delta T_1 = \frac{[1 - a(T_0)]\delta g}{B(T_0)}.$$

If the CO₂ level in the atmosphere doubles, then the radiative forcing might be adjusted as:

$$(1 - a(T_0))\delta = 3.7 \text{ W} \cdot \text{m}^{-2}.$$

Assuming that the climate gain is $g = 3$ and $B(T_0) = 1.9 \text{ W} \cdot \text{m}^{-2} \cdot (\text{ }^\circ\text{C})^{-1}$, what is the expected increase in temperature?

CHAPTER 30

Problem set 6

Note

Note that the material for PS6 will not be covered until after the Easter break; both PS6 and PS7 are the last problem sets of the term!

Q1. A van der Pol equation

Consider the van der Pol equation for $x(t)$ with $0 < \epsilon \ll 1$:

$$\epsilon \frac{d^2x}{dt^2} + (3x^2 - 6x - 9) \frac{dx}{dt} + 4x = 0.$$

- Determine $f(x)$ so that this equation can be written as a Liénard phase plane system in the form

$$\begin{aligned}\epsilon \dot{x} &= f(x) + 4y, \\ \dot{y} &= -x.\end{aligned}$$

- For fixed $\epsilon > 0$, find the equilibrium point(s) in the phase plane, find their eigenvalues, and classify their linear stability.
- Use the expansions $x(t) = x_0(t) + \epsilon x_1(t) + O(\epsilon^2)$ to determine the equations for the leading-order slow solution. Sketch the slow manifold, indicate the directions of motion on each part, and identify the two attracting points on the curve.
- Use the expansions $x(t) = X_0(T) + \epsilon X_1(T) + O(\epsilon^2)$ and $y(t) = Y_0(T) + \epsilon Y_1(T) + O(\epsilon^2)$ with $T = t/\epsilon$ to obtain the leading-order fast solution.
- Use the phase plane to determine the maximum and minimum values of $x(t)$ during an oscillation. Sketch $x(t)$ as a function of time.

Q2. Fast-slow dynamics with three variables

Consider the limit of $\epsilon \ll 1$ for each of the following cases.

a. Consider the system

$$\begin{aligned}\dot{x} &= 2 - y, \\ \dot{y} &= x - z, \\ \epsilon \dot{z} &= y - y^2 + \frac{1}{3}y^3 - z,\end{aligned}$$

with the initial conditions of $x(0) = 1$, $y(0) = 3$, and $z(0) = 0$.

- (i) Identify the surface $z = S(x, y)$ that defines the slow manifold.
- (ii) Find the equilibrium point of the leading-order slow phase plane system and show that it is asymptotically stable for $t \rightarrow \infty$.
- (iii) Also determine the form of the initial layer that describes the transition from the initial conditions to the slow manifold.

b. Consider the system

$$\begin{aligned}\dot{x} &= 2 - y, \\ \epsilon \dot{y} &= x - z, \\ \dot{z} &= y - y^2 + \frac{1}{3}y^3 - z,\end{aligned}$$

with the initial conditions of $x(0) = 0$, $y(0) = 3$, and $z(0) = 1$.

- (i) Show that the slow manifold reduces to a curve that could be written in parametric form as $x = x(z)$, $y = y(z)$, $z = z$.
- (ii) Determine the asymptotic solution for $t \rightarrow \infty$.
- (iii) Also determine the form of the initial layer that describes the transition from initial conditions to the slow manifold.

Q3. Visualising fast-slow dynamics

By modifying the standard scripts developed in lectures, plot visualisations of Q1 and Q2(a) in order to verify your understanding of the fast-slow dynamics.

CHAPTER 31

Problem set 7

Q1. Non-dimensionalisation of the Stommel box model

Consider the Stommel box model given in (Equation 23.1). By following the procedure outlined in Section 23.1 non-dimensionalise the model in order to produce the set of equations given in (Equation 23.2), repeated here:

$$\begin{aligned}\frac{dx}{dt} &= \delta(1 - x) - |f(x, y)|x, \\ \frac{dy}{dt} &= 1 - y - |f(x, y)|y,\end{aligned}$$

where we have introduced the function,

$$f(x, y; R, \lambda) = \frac{1}{\lambda}(Rx - y).$$

Provide a brief description of the physical interpretations of the parameters λ , δ , and R .

Q2. Investigations for R

Using the script available on Noteable, investigate the behaviour of the system in the above question under changing values of R and fixed values of the other parameters. Based on your intuition, what do you expect to be the features of the bifurcation diagram, as shown in the (R, f) -plane?

Q3. Alternative scalings of the Stommel box model

It is possible to scale the problem differently, and this may allow for simpler analysis. Consider the set of equations in Q2. Write instead

$$x = [x]s, \quad y = [y]\theta, \quad t = [t]t',$$

Choose the correct scalings, $[x]$, $[y]$, and $[t]$, so that we obtain a re-scaled version:

$$\begin{aligned}\frac{ds}{dt'} &= 1 - (\epsilon + |q|)s, \\ \frac{d\theta}{dt'} &= 1 - (\mu + |q|)\theta,\end{aligned}$$

where now the flow term is given by

$$q = \kappa(-\theta + \tilde{R}s).$$

In addition to the choice of scalings, you will need to describe how the new parameters, ϵ , μ , κ , and \tilde{R} , are related to the older parameters.

Q4. Box models for flooding 1

Consider the basic flooding model given in (Equation 24.1), and repeated below:

$$\frac{dS}{dt} = P(t) - P(t)H(S - S_{\max}) - E_p(t)\frac{S}{S_{\max}}.$$

Consider the initial condition of $S(0) = S_0 \geq 0$, and a constant rainfall $P(t) = P$ and constant potential evapotranspiration $E_p(t) = E_p$. Describe, using analysis, the evolution of the system.

Draw graphs of the storage, S , and saturation excess runoff Q_{se} (the second term of the above equation).

CHAPTER 32

Problem set 1 solutions

Q1 The bump lemma

Prove the following one-dimensional lemma, which was used in the derivation of the heat equation.

If $\int_a^b g(x) dx = 0$ for all a and $b \in [0, 1]$, then $g(x) \equiv 0$ for all $x \in [0, 1]$.

Hint: think of a proof by contradiction.

Solution

Assume by contradiction that $f(x) \neq 0$ between two arbitrary points, say x_0 and x_1 , both in the interval $[a, b]$. Then $\int_{x_0}^{x_1} f(x) dx \neq 0$. This assumes some degree of smoothness and integrability (as long as f is not too pathological).

Q2. A source in the heat equation

Consider the same heat experiment discussed in Chapter 2 but now consider a bar that has an internal source or sink generating or removing heat, such as the case of a boiler with an internal heating element. By adapting a similar derivation to the one presented, explain why the modified conservation of heat equation is

$$\frac{\partial}{\partial t} \int_a^b \rho c T dx = q(x = a, t) - q(x = b, t) + \int_a^b R(x, t) dx.$$

In addition:

- By studying the dimensions of the other terms in the above equation, find what the dimensions of R are. What does $R > 0$ mean and $R < 0$?

Solution

On the left, density has units kg/m^3 , specific heat $\text{J}/(\text{kg K})$, and temperature K . But then dx has units of m . And d/dt has units of $1/\text{s}$. So altogether, we have the

following SI units on the left:

$$\frac{J}{m^2 \cdot s}$$

So its energy per unit area per unit time. The reason why it's per unit area is because you have not integrated over the sides of the cylinder/bar.

So therefore this needs to match the units on the right, which are $[R] [dx]$. So you conclude that

$$[R] = \frac{J}{m^3 \cdot s}$$

i.e. energy per unit time per unit volume. This is energy density.

$R > 0$ means heat is added into the system (source). $R < 0$ means heat is removed from the system (sink).

- b. Hence derive the partial differential equation that governs the temperature T .

Solution

The derivation follows exactly as in the notes, except you have an additional term on the RHS. Use Fourier's law to convert $q = -k\partial T/\partial x$. Write everything under the integral and this yields:

$$\rho c \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} + R.$$

- c. By introducing the appropriate scalings on each of the variables, x , t , and T , non-dimensionalise the PDE and discuss the non-dimensional parameters (there will be two).

Solution

Same derivation as in the notes. Let

$$x = Lx', T = [T]T' \quad \text{and} \quad t = [t]t'.$$

This time you will remember to scale out the typical source strength, so

$$R = [R]R'.$$

This leads to

$$\frac{\partial T'}{\partial t'} = \left(\frac{k[t]}{L^2 \rho c} \right) \frac{\partial^2 T'}{\partial x'^2} + \left(\frac{[R][t]}{\rho c[T]} \right) R'.$$

So this gives the two sets of non-dimensional numbers

$$\Pi_1 = \frac{k[t]}{L^2 \rho c} \tag{32.1}$$

$$\Pi_2 = \frac{[R][t]}{\rho c[T]} \tag{32.2}$$

If you want, you can write it in the alternative form by taking the time scaling to be $[t] = L/U$, where U is some unspecified velocity scale.

For the interpretation, you can simply take Π_1 to be a measure of diffusive effects (heat spreading out) and Π_2 to be a measure of source strength. Note that this problem can be simplified even further by choosing the time scale so as to set $\Pi_1 = 1$. If you do this, you will be left with a single non-dimensional parameter that balances diffusion with the strength of the source.

Q3. The unique timescale in the heat equation

During our investigation of the heat equation, we found that it was possible to scale time so as to scale out the only non-dimensional parameter that appears in the PDE. As explained in lectures, the disappearance of all non-dimensional parameters is due to the fact that only a single sensible timescale exists.

By adjusting the boundary conditions, we may create a new problem involving heat flow where a unique ‘special’ timescale can no longer be chosen.

Consider a system where one side of the rod is heated in some periodic fashion, e.g. set the initial and boundary conditions to be

$$T(x, 0) = T_0 \quad (32.3)$$

$$T(0, t) = T_a \cos(\omega t), \quad (32.4)$$

$$T(L, t) = T_b. \quad (32.5)$$

- a. What must the units of ω be?

Solution

Note ωt must be unitless. Therefore $[\omega] = 1/s$.

- b. Non-dimensionalise as usual and, without selecting the timescale, $[t]$, identify the key non-dimensional parameters that remain. Write a brief sentence to describe their physical interpretation.

Solution

This is exactly the same as in the above heat question. You will still obtain

$$\frac{\partial T'}{\partial t'} = \Pi_1 \frac{\partial^2 T'}{\partial x'^2}$$

where the non-dimensional parameters is the Peclet number introduced in lectures:

$$\Pi_1 = \left(\frac{k[t]}{L^2 \rho c} \right).$$

The main difference is that now, your boundary conditions are

$$T'(0, t') = \Pi_3 \cos(\Pi_2 t') \quad \text{and} \quad T'(1, t') = \Pi_4$$

where

$$\Pi_2 = \omega[t], \quad \Pi_3 = \frac{T_a}{T_0}, \quad Pi_4 = T_b/T_0.$$

The scaling on the temperature was taken to be such that the final initial condition is set to unity:

$$T'(x', 0) = 1.$$

- c. There are two sensible choices for setting the timescale, $[t]$. Identify the two choices and present the reduced set of equations in each case.

Solution

So there are now two ways of choosing the time scale. Either choose it to set $\Pi_1 = 1$ or choose it to set $\Pi_2 = 1$. In the former choice, you are choosing to set time according to the diffusive rate. In the latter, you are setting time according to how the boundary is heated.

Q4. Timescale in the surface energy

Take the basic zero-dimensional energy model studied in (Equation 4.7) for the temperature of the troposphere:

$$c \frac{dT}{dt} = \frac{1}{4} Q(1-a) - \sigma\gamma T^4.$$

- a. Non-dimensionalise the model by choosing $T = [T]T'$, $t = [t]t'$, and $Q = [Q]Q'$. Show that it is possible to select the scalings on the temperature and time so as to completely remove all constants from the problem when Q is assumed to be constant.

Solution

The question's suggestion of setting $Q = [Q]Q'$ was excessive since Q is just a constant in this case. So proceeding as usual:

Set in the scalings. This time it is easier to balance the right hand-side first.

$$\frac{c[T]}{[t']} \frac{\partial T'}{\partial t'} = \left(\frac{Q(1-a)}{4} \right) - (\sigma\gamma[T]^4) T'^4.$$

So select the temperature scaling so that both coefficients on the RHS are equal.

$$[T] = \left(\frac{Q(1-a)}{4\sigma\gamma} \right)^{1/4}.$$

Now choose the time scaling also to match the LHS coefficient to the RHS coefficient.

$$c \frac{[T]}{[t]} = \sigma\gamma[T]^4 \implies [t] = \frac{c}{\sigma\gamma[T]^4}$$

Once you have done this, the equation transforms to the one given in part b.

- b. Thus, show that the analysis of the above equation is equivalent to studying

$$\frac{dT}{dt} = 1 - T^4,$$

where we have dropped the primes and assumed that the albedo is such that $1-a \neq 0$.

Solution

Done above.

- c. From your choice of $[t]$, estimate the typical dimensional value using $d \approx 10\text{km}$, $\rho \approx 1\text{kg m}^{-3}$, $c_p \approx 10^3\text{J kg}^{-1}\text{K}^{-1}$.

Use a pocket calculator to verify your calculations and conclude that this time-scale is on the order of a month. What is the relevance of this approximation as it concerns the steady-state solution?

Solution

Below (Equation 4.7), we defined $C = \rho c_p d$. So the scaling on time is (put in the scaling for [T]):

$$[t] = \left(\frac{4}{(1-a)} \right)^{3/4} \left(\frac{C}{(\sigma\gamma)^{1/4} Q^{3/4}} \right).$$

So we know that $\sigma = 5.67 \times 10^{-8} \text{ J/(m}^2\text{s K}^4)$, $Q = 1.38 \times 10^3 \text{ J/(m}^2\text{s)}$. We can also calculate

$$C = \rho c_p d = 10^7 \text{ J/(K m}^2).$$

We can use $\gamma = 1$ for an estimate. The right-most factor is

$$\frac{10^7}{(5.67 \times 10^{-8})^{1/4} (1.38 \times 10^3)^{3/4}} s = 2.88 \times 10^6 s$$

If we use $a = 0.3$ then this gives a total of

$$[t] = 3.7 \times 2.88 \times 10^6 s.$$

To convert to days, note $1 \text{ day} = 24 \text{ hours} = 2460 \text{ minutes} = 2460 * 60 \text{ seconds}$. This is then

$$[t] = \frac{1.65 \times 10^7}{8.64 \times 10^4} \text{ days} = 0.19 \times 10^3 \text{ days} = 190 \text{ days.}$$

So on the order of 6 months. Any answer you get that's roughly on the order of months sounds about right. This is the rough approximation of how long time-scale effects take to appear.

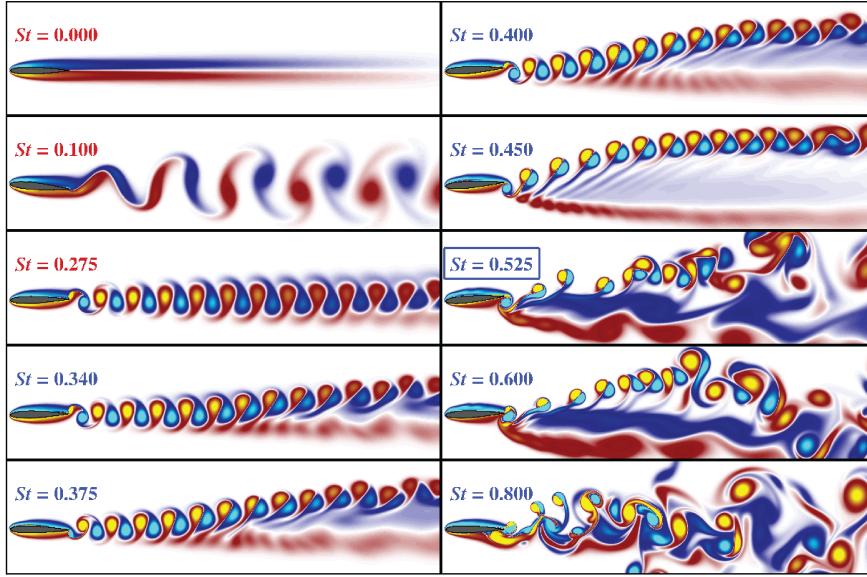
Q5. Navier-Stokes Equation

Suppose that the time, with $t = [t]t'$ is non-dimensionalised as $[t] = \frac{1}{f}$ where f is a frequency. Show that the Navier-Stokes equations, introduced in Problem Class 2, can be written as

$$\frac{\partial u}{\partial t} + \frac{1}{St} (u \cdot \nabla) u = -\nabla p + \frac{1}{Re} \nabla^2 u.$$

Above, the Reynolds number is defined via $Re = \frac{f[x]^2}{\nu}$ and the Strouhal number is defined via $St = \frac{[x]f}{[u]}$. The Strouhal number represents the effect of vortex shedding

in flow across an object. The higher St is, the more vortices will be generated by the flow.



Solution

After introducing the non-dimensional terms and making the coefficient of the time derivative equal to 1, the Navier-Stokes equation becomes

$$\frac{\partial u}{\partial t} + \frac{[u][t]}{[x]} (u \cdot \nabla) u = -\frac{[p][t]}{[u][x]\rho} \nabla p + \frac{\nu[t]}{[x]^2} \nabla^2 u.$$

If $[t] = \frac{1}{f}$ and $\frac{[p][t]}{[u][x]\rho} = 1$ (meaning that $[p] = \frac{[u][x]\rho}{[t]} = [u][x]f\rho$), then

$$\frac{\partial u}{\partial t} + \frac{[u]}{[x]f} (u \cdot \nabla) u = -\nabla p + \frac{\nu}{f[x]^2} \nabla^2 u.$$

Let $\text{Re} = \frac{f[x]^2}{\nu}$ and $\text{St} = \frac{[x]f}{[u]}$, then the Navier-Stokes equation can be written as

$$\frac{\partial u}{\partial t} + \frac{1}{\text{St}} (u \cdot \nabla) u = -\nabla p + \frac{1}{\text{Re}} \nabla^2 u.$$

as required.

By considering the SI units for the different quantities of f , L , ν , and U , confirm that the two non-dimensional numbers introduced above are indeed non-dimensional.

Solution

$$[\text{Re}] = [f][x]^2[\nu]^{-1} = (\text{s}^{-1}) (\text{m})^2 (\text{m}^2 \text{s}^{-1}) = 1.$$

$$[\text{St}] = [x][f][u]^{-1} = (\text{m}) (\text{s}^{-1}) (\text{m s}^{-1})^{-1} = 1.$$

CHAPTER 33

Problem set 2 solutions

33.1 Getting started with Jupyter

Note

N/A

33.2 Testing the solutions of a cubic

Note

See `notebook/ps2_sol.ipynb` notebook for the code.

33.3 Analysis of singular cubic equation

Consider the cubic equation

$$\epsilon x^3 - x + 1 = 0,$$

with $\epsilon \ll 1$ and $\epsilon > 0$.

1. Develop the first three terms of an asymptotic expansion about the root by setting

$$x = x_0 + \epsilon x_1 + \epsilon^2 x_2 + \dots$$

Solution

Substitution into the cubic gives

$$\epsilon(x_0^3 + 3x_0x_1 + \dots) - (x_0 + \epsilon x_1 + \epsilon^2 x_2) + 1 = 0.$$

Equating at each order and solving gives

$$-x_0 + 1 = 0 \implies x_0 = 1 \quad (33.1)$$

$$x_0^3 - x_1 = 0 \implies x_1 = 1 \quad (33.2)$$

$$3x_0x_1 - x_2 = 0 \implies x_2 = 3 \quad (33.3)$$

So the three-term approximation is

$$x \sim 1 + \epsilon + 3\epsilon^2.$$

2. Fill out the following table.

Solution

$ \epsilon $	x_{exact}	$ x_{\text{exact}} - x_0 $
0.1	1.1535	0.1535
0.08	1.1092	0.1092
0.06	1.0744	0.0744
0.04	1.0457	0.0457
0.02	1.0213	0.0213

This is a really excellent demonstration of an ‘organic’ discovery process. Below we plot the errors we found in the above table. I rounded the values to only the first two decimals. It is not so important to be extremely accurate (in fact, this is the point of requiring you to do so by hand!)

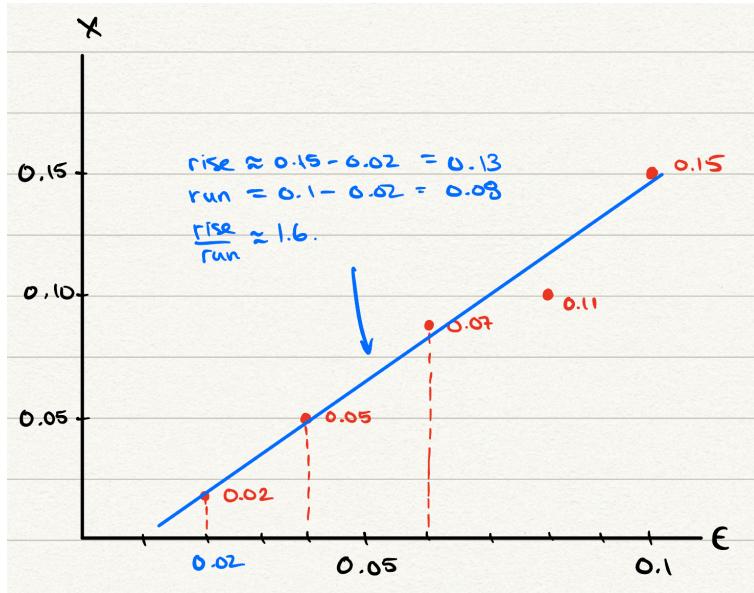


Figure 33.1: Hand drawn image

When doing this, I tried to fit to a line because I knew the error was supposed to be

$$x - x_0 \sim \epsilon$$

so was expecting a line of unit gradient. But the gradient estimated above was a bit higher, at 1.6. Returning to Python, we see the problem.

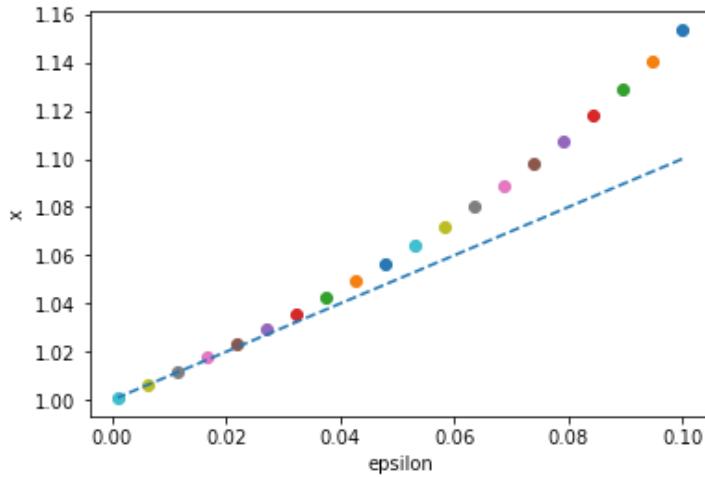


Figure 33.2: Python image

So over the range investigated of $0.02 < \epsilon < 0.1$, the behaviour is quadratic. This is a nice lesson on how to discover new results based on computation and analysis.

3. By rescaling x appropriately in terms of ϵ , derive the first three terms of the asymptotic approximations of the remaining roots.

Solution

We expect the critical scaling to balance ϵx^3 with x since these two will be dominant when x is large. So we set

$$x = \frac{X}{\sqrt{\epsilon}}$$

Now we have

$$X^3 - X + \sqrt{\epsilon} = 0.$$

You will notice that the progression here should go in powers of $\epsilon^{1/2}$. Alternatively set $\delta = \epsilon^{1/2}$ and now the problem is standard in δ :

$$X^3 - X + \delta = 0.$$

And we expand

$$X = X_0 + \delta X_1 + \delta^2 X_2 + \dots$$

We get

$$X_0^3 - X_0 = 0 \implies X_0^2 = 1 \implies X_0 = \pm 1 \quad (33.4)$$

$$3X_0^2 X_1 - X_1 + 1 = 0 \implies X_1 = -\frac{1}{2} \quad (33.5)$$

$$3X_0^2 X_2 + 3X_0 X_1^2 - X_2 = 0 \implies X_2 = \mp \frac{3}{8}. \quad (33.6)$$

So it seems that the three-term approximation for the other two roots is

$$x \sim \frac{1}{\sqrt{\epsilon}} \left(\pm 1 - \sqrt{\epsilon} \frac{1}{2} \mp \epsilon \frac{3}{8} \right).$$

33.4 A damped projectile problem

If air resistance is included, then the non-dimensional toy model is instead

$$\frac{d^2y}{dt^2} = -\frac{1}{(1+\epsilon y)^2} - \frac{\alpha}{(1+\epsilon y)} \frac{dy}{dt}, \quad (33.7)$$

$$y(0) = 0, \quad (33.8)$$

$$y'(0) = 1. \quad (33.9)$$

where $\alpha \geq 0$ is the parameter that controls air resistance.

1. Begin by assuming that α is a fixed number and consider the limit where $\epsilon \ll 1$. Find a one-term asymptotic expansion of the solution for small ϵ .

Solution

First, note that

$$(1+\epsilon y)^\alpha = 1 + \alpha(\epsilon y) + O(\epsilon y)^2.$$

You will not need more terms than that. Substitute $y = y_0(t) + \epsilon y_1(t) + \dots$ into the equation. This gives, for the leading terms

$$y_0'' = -1 - \alpha y_0' \quad (33.10)$$

$$y_0(0) = 0 \quad (33.11)$$

$$y_0'(0) = 1 \quad (33.12)$$

To solve this, you will need to refresh your memory on second-order differential equations learned in 2nd year. You can see Chapter 45 for references. To solve this set $y_0 = e^{rt}$ and solve for r for the homogeneous equation. You will get

$$y_{\text{homogeneous}} = (c_1 + c_2 e^{-\alpha t}).$$

From here, we need a particular solution. This can be obtained by guessing (also see [this reference](#)). If we guess

$$y_p = Ct,$$

we see that

$$y_p'' + \alpha y_p' = \alpha C = -1$$

implies $C = -1/\alpha$. Therefore the general solution is

$$y_0 = (c_1 + c_2 e^{-\alpha t}) - \frac{t}{\alpha}.$$

We now select the constants to satisfy the initial conditions. This yields

$$c_1 + c_2 = 0 \quad \text{and} \quad -\alpha c_2 - 1/\alpha = 1,$$

from which the solution can be easily derived. It is

$$y(t) \sim \left(\frac{1}{\alpha^2} + \frac{1}{\alpha} \right) - \left(\frac{1}{\alpha^2} + \frac{1}{\alpha} \right) e^{-\alpha t} - \frac{t}{\alpha}.$$

The solution of y_1 proceeds in the same way, but since we did not anticipate that this question would be so algebraically involved, we will not require solving for y_1 . This can be left as an exercise if you so wish.

2. (Challenging) Is the effect of the air resistance to increase or decrease the flight time? Justify based on your analytical solution.

Solution

The above solution seems quite strange. After all, when $\alpha = 0$, we know that parabolic motion is expected. We found previously that

$$y_{\alpha=0}(t) \sim -\frac{1}{2}t^2 + t.$$

which matches our intuition about the expected flight being parabolic. However, notice that if α is small, we can expand the exponential, $e^{-\alpha t} = 1 - \alpha t + \dots$. Substitution into the derived formula for y_0 gives

$$y(t) \sim \left(\frac{1}{\alpha^2} + \frac{1}{\alpha}\right) - \left(\frac{1}{\alpha^2} + \frac{1}{\alpha}\right)[1 - \alpha t + \dots] - \frac{t}{\alpha}.$$

We can now simplify the expression and we see that the $\alpha \rightarrow 0$ limit is not problematic, because it now becomes:

$$y_0 \sim \left(-\frac{1}{2}t^2 + t\right) + \alpha \frac{1}{6}(-3t^2 + t^3) + O(\alpha^2),$$

and the previous terms that we believed might tend to infinity as $\alpha \rightarrow 0$ cancel. Inspecting the above solution, notice that the leading-order result agrees with the previously derived result for $\alpha = 0$. The next term shows the small effect of drag. You can simply plot the curves and show the difference.

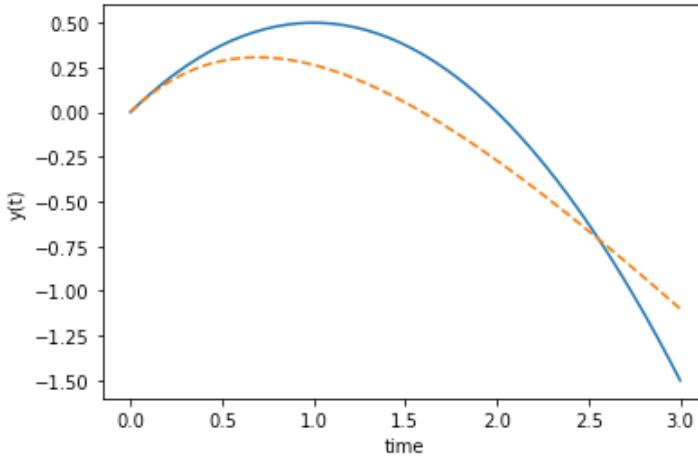


Figure 33.3: With (solid) and without damping (dashed)

Notice that the dashed line brings down the parabolic path initially.

33.5 ODE solvers and Euler's method

Return to the setup of the above question.

1. Modify the script shown in Section 6.2 in order to solve the equation from the previous question at a prescribed value of ϵ and α .

Solution

This is found in `problemsets/sol02_solutions.ipynb`.

- Using a pocket calculator (or your phone calculator) apply Euler's method with $\Delta t = 0.2$, $\epsilon = 0.2$, and $\alpha = 1$ to determine the position of the projectile at $t = 0.6$.

Solution

Euler's method is simply

$$Y_{n+1} = Y_n + F(t_n, y_n)\Delta t.$$

Here

$$F = \begin{pmatrix} y' \\ -\frac{1}{(1+\epsilon y)^2} - \frac{\alpha}{(1+\epsilon y)} \end{pmatrix}.$$

We split the time steps into $\{0, 0.2, 0.4, 0.6\}$. We also round to two figures.

t	Y	$1 + \epsilon y$	F	$F\Delta t$
0	(0, 1)	1	(1, -2)	(0.2, -0.4)
0.2	(0.2, 0.6)	1.04	(0.6, -1.5)	(0.12, -0.3)
0.4	(0.32, 0.3)	1.06	(0.3, -1.16)	(0.06, -0.23)
0.6	(0.38, 0.07)			

After three iterations, we get a value of $y(0.6) \approx 0.38$.

- Compare your hand calculation with the result from the Python output, as well as with your asymptotic approximations.

Solution

We have (rounded to two digits)

$$y_{\text{exact numerical}} \approx 0.31$$

$$y_{\text{euler}} \approx 0.38$$

$$y_{\text{asym}} \approx 0.30$$

CHAPTER 34

Problem set 3 solutions

$$\begin{aligned}\epsilon T'' + 2T' + T &= 0, \\ T(0) &= 0, \\ T(1) &= 1.\end{aligned}$$

The challenge is to study the above problem for small values of ϵ .

34.1 Conversion to first-order system

By using the procedure reviewed in Section 6.2, convert (Equation 27.1) to a first-order system of equations.

Solution

Introduce $u(z) = T(z)$ and $v(z) = T'(z)$. Then

$$\frac{d}{dz} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} v \\ -\frac{1}{\epsilon}(2v + u) \end{pmatrix}$$

34.2 Numerical solutions

Solution

A script can be found in `problemsets/ps03_solutions.ipynb`

34.3 Investigation of the boundary layer

Using your code, use the following command in order to investigate the maxima (x_m, T_m) as ϵ varies:

```
ind = np.argmax(Y[0])
print(ep, z[ind], Y[0, ind])
```

You may want to consider, as an example, the values $\epsilon = \{0.05, 0.1, 0.15, 0.2\}$ and fill the following table.

ϵ	z_m	T_m
0.05	0.1122	1.5479
0.1	0.1924	1.4742
0.15	0.2605	1.4094
0.2	0.3226	1.3501

Create a plot of (ϵ, x_m) and discuss the observed trend and its implications.

Solution

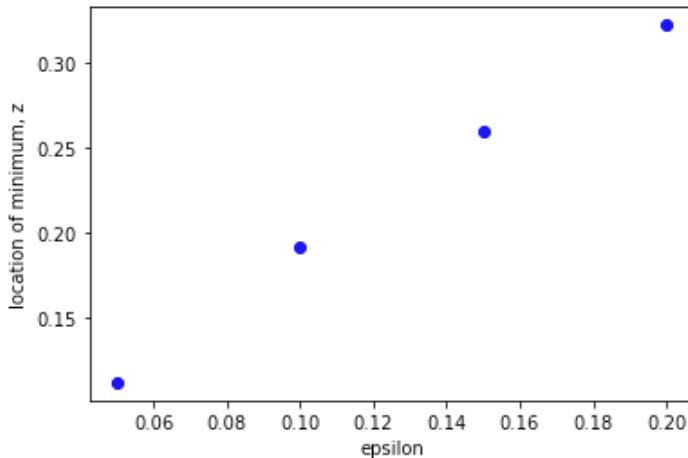


Figure 34.1: A plot of the above data. This looks linear, and indeed this provides numerical confirmation that the boundary layer should be sized with ϵ

34.4 Outer asymptotic solutions

Begin by setting

$$T = T_{\text{outer}} = T_0(z) + \epsilon T_1(z) + \epsilon^2 T_2(z) + \dots$$

Substitute the above expansion into the system and solve for the first two orders.

You may verify that the solution is given by

$$T_0 = e^{1/2} e^{-z/2}, \quad (34.1)$$

$$T_1 = -\frac{1}{8} e^{1/2} e^{-z/2} (z - 1). \quad (34.2)$$

Solution

Remember: as you solve the outer solution, you impose the boundary condition of $T(1) = 1$.

Leading-order: at leading order, we get the problem $2T'_0 + T_0 = 0$ subject to $T_0(1) = 1$. This is solved either by separation of variables or by integrating factor. Using the technique of solving first-order linear equations, (see Chapter 45) we

divide,

$$T'_0 + \frac{1}{2}T_0 = 0.$$

We then multiply both sides of the equation by $e^{z/2}$.

$$\frac{d}{dz} [T_0 e^{z/2}] = 0.$$

Integrate, and solve, giving $T_0 = Ce^{-z/2}$. We now impose $T_0(1) = 1$ which yields $C = e^{1/2}$.

First-order: at next order, we get

$$2T'_1 + T_1 = -T''_0 \implies T'_1 + \frac{1}{2}T_1 = -\frac{1}{2}T''_0,$$

along with the boundary conditions of $T_1(1) = 0$.

Again place this into the appropriate form. Again, this is a linear first-order equation. We multiply by the integrating factor $e^{z/2}$, placing the equation into the form

$$\frac{d}{dz} [T_1 e^{z/2}] = -\frac{1}{2}T''_0 e^{z/2} = -\frac{1}{8}e^{1/2}.$$

We integrate once and simplify, yielding

$$T_1(z) = \left(-\frac{e^{1/2}}{8}z + C \right) e^{-z/2},$$

and then impose the condition that $T_1(1) = 0$. This yields $C = e^{1/2}/8$ hence

$$T_1(z) = -\frac{e^{1/2}}{8}(z-1)e^{-z/2}.$$

34.5 Inner asymptotic solutions

There will be a boundary layer near $z = 0$. Set $z = \epsilon^\alpha s$ and $T(z) = U(s)$. Follow the same procedure, as in Chapter 8 in order to determine the correct choice of α for the inner region. This choice should balance the two terms $\epsilon T''$ and $2T'$.

Solution

We re-scale $z = \epsilon^\alpha s$ and let $T(s) = U(s)$. Then the ODE becomes

$$\epsilon^{1-2\alpha}U'' + 2\epsilon^{-\alpha}U' + U = 0.$$

Balancing the first two terms thus requires that

$$1 - 2\alpha = -\alpha \implies \alpha = 1.$$

Thus the differential equation now becomes

$$U'' + 2U' + \epsilon U = 0.$$

34.6 Matching and comparison

Expanding the inner solution as $U = U_0(s) + \epsilon U_1(s) + \dots$, write down the equation and boundary conditions that U_0 must satisfy. You will notice that U_0 is governed by a second-order differential equation and therefore needs two boundary conditions. One boundary condition comes from $z = 0$, i.e.

$$U_0(0) = 0.$$

The other boundary condition is a matching condition:

$$\lim_{s \rightarrow \infty} U_0(s) = \lim_{z \rightarrow 0} T_0(z),$$

which imposes that the inner solution, as it leaves the boundary layer, matches the outer solution, it tends into the inner region.

Solution

Solve for U_0 .

We thus expand $U = U_0 + \epsilon U_1 + \dots$. To leading order, we have

$$U_0'' + 2U_0' = 0.$$

Integrating, we find that the general solution is given by

$$U_0(s) = C_1 + C_2 e^{-2s}.$$

We require two boundary conditions. The first condition is given by $T(0) = 0$ and hence $U(0) = 0$. The second condition is given by requiring that the inner solution, as $s \rightarrow \infty$, matches the outer solution, as $z \rightarrow 0$. Using our solution above, we find that

$$\lim_{z \rightarrow 0} T_0(z) = e^{1/2}.$$

Thus the second boundary condition is

$$\lim_{s \rightarrow \infty} U_0(s) = e^{1/2}.$$

Together, both boundary conditions are used to conclude that

$$U_0(s) = e^{1/2}(1 - e^{-2s}),$$

or in terms of z , the inner solution is

$$T_{\text{inner}} \sim e^{1/2}(1 - e^{-2z/\epsilon}).$$

CHAPTER 35

Problem set 4 solutions

Note

Note that Q1 and Q4 are new to complement the coverage on numerical solutions of PDEs.

Q1. Finite difference formulae

Solution a)

- Show that

$$U(x+h) = U(x) + hU'(x) + \frac{h^2}{2}U''(x) + \frac{h^3}{6}U'''(x) + \dots,$$
$$U(x-h) = U(x) - hU'(x) + \frac{h^2}{2}U''(x) - \frac{h^3}{6}U'''(x) + \dots.$$

This is straightforward and by definition of Taylor's theorem.

Solution b)

- By subtracting the two equations above, derive the centred difference formula used for the first-derivatives in our algorithms:

$$U'(x) = \frac{U(x+h) - U(x-h)}{2h} + O(h^2).$$

This is what's known as the centred difference formula for the derivative.

Again this is immediate by subtracting the given formulae.

Solution c)

- By adding the two Taylor series equations above, derive the centred differ-

ence formula used for the second-derivatives in our algorithms:

$$U''(x) = \frac{U(x+h) - 2U(x) + U(x-h)}{h^2} + O(h^2).$$

This part is also immediate. Adding the two equations gives

$$U(x+h) + U(x-h) = 2U(x) + h^2 U''(x) + O(h^3),$$

and simplify.

Solution d)

- d. The centered difference formula for $U'(x)$ will not immediately work if it is applied to find the derivative at the first mesh point in a discretisation, e.g. $x = x_0$. Show that the forward difference formula at $x = x_0$ can be derived as

$$U'(x_0) = \frac{-3U(x_0) + 4U(x_1) - U(x_2)}{2h} + O(h^2).$$

Hint: consider as well the Taylor series for $U(x+2h)$ and then combine the expansions for $U(x+h)$ and $U(x+2h)$ to derive the result.

This is a bit more involved. Let us explain firstly why the standard approach doesn't work:

$$\begin{aligned} U(x+2h) &= U(x) + 2hU'(x) + 2h^2U''(x) + \frac{4}{3}h^3U'''(x) + \dots \\ U(x+h) &= U(x) + hU'(x) + \frac{h^2}{2}U''(x) + \frac{h^3}{6}U'''(x) + \dots \end{aligned}$$

Now adding gives

$$U(x+2h) + U(x+h) = 2U(x) + 3hU'(x) + \frac{5h^2}{2}U''(x) + \dots$$

However, if we isolate $U'(x)$ we obtain a formula where the error term is $O(hU''(x))$ instead of $O(h^2)$.

The solution is to consider a linear combination:

$$U'(x) = aU(x) + bU(x+h) + cU(x+2h),$$

and insert the above formulae. Keeping track of the term multiplying the second derivative, $U''(x)$, we then need:

$$b/2 + 2c = 0 \implies b = -4c.$$

The remaining two equations are:

$$a + b + c = 0, \quad bh + 2ch = 1.$$

The second yields $c = -1/(2h)$ and hence $b = 2/h$. The first equation then gives $a = -3/(2h)$.

Q2. The wine cellar problem I

Solution

Separation of variables leads to the equation

$$\frac{G'}{G} = \kappa \frac{H''}{H} = \lambda.$$

Crucially in considering possible solutions, we want solutions to obey the necessary boundary condition of $T(0, t) = A \cos(\omega t) = G(t)H(0)$. This is highly restrictive. We first consider $\lambda = 0$. Then we have that $G(t) = \text{const.}$. This is then no way to satisfy the time-varying boundary condition.

We next consider $\lambda \in \mathbb{R}$. Then $G(t) = Ce^{\lambda t}$. So again the behaviour of the time-dependent part is incompatible with the oscillating boundary conditions. Therefore the only remaining possibility is $\lambda = i\alpha$ where α is real. Then we have that

$$G(t) = C_1 e^{i\alpha t} + C_2 e^{-i\alpha t}.$$

Written in terms of real-valued functions, this gives essentially a linear combination of $\cos(\alpha t)$ and $\sin(\alpha t)$. In order to be compatible with the boundary conditions, we would thus need

$$G(t) = \text{const.} \cos(\omega t) = \text{const.} \times \operatorname{Re}(e^{i\omega t}),$$

i.e. taking the imaginary part (if we wish to continue manipulating exponentials). Now solving for $H(x)$, we set $H(x) = e^{rx}$ as a solution ansatz, giving the characteristic equation

$$r^2 = i\omega \implies r = \pm\sqrt{i\omega}.$$

Therefore the general solution for H yields

$$H(x) = D_1 e^{\sqrt{i\omega}x} + D_2 e^{-\sqrt{i\omega}x}.$$

In order to parse the above form, it is best to re-write the real and complex parts of the exponential argument separately. We note that by Euler's theorem,

$$\sqrt{i} = (e^{\pi i/2})^{1/2} = e^{\pi i/4} = \frac{1+i}{\sqrt{2}}.$$

So now we can re-write

$$H(x) = D_1 e^{\sqrt{\omega/2}x} e^{i\sqrt{\omega/2}\kappa x} + D_2 e^{-\sqrt{\omega/2}\kappa x} e^{-i\sqrt{\omega/2}x}.$$

By the boundedness assumption, we require that $H(x)$ does not blow up as $x \rightarrow \infty$. So therefore $D_1 = 0$. We can finally put the results together, yielding:

$$T(x, t) = \text{const.} \times \operatorname{Re} [e^{i\omega t} e^{-\sqrt{\omega/2}\kappa x} e^{-i\sqrt{\omega/2}\kappa x}] = \text{const.} \times \operatorname{Re} [e^{-\sqrt{\omega/2}\kappa x} e^{i(-\sqrt{\omega/2}\kappa x + \omega t)}].$$

Note that the reason you have re-written it with the exponentials arranged in this fashion is because the portion of the exponential with the imaginary argument becomes oscillatory. Taking the real part yields

$$T(x, t) = A e^{-\sqrt{\omega/2}x} \cos(-\sqrt{\omega/2}\kappa x + \omega t).$$

You can double check that the above solution satisfies all the required conditions, including the condition at $x = 0$ (forcing us to choose the prefactor A).

Q3. The wine cellar problem II

Solution

Solution script can be found on the solutions/ folder of the Noteable Python website.

The numerical solution is tricky for two reasons. First, you must consider the right range of parameters. Second, the numerical algorithm suffers from instability, so you must consider the right spatial and time discretisation.

To begin, from the notes Chapter 12, you are given that typical parameters are

$$\omega = \frac{2\pi}{3.15 \times 10^7 \text{ s}}.$$

This has already been converted to seconds, so instead use

$$\omega = 2\pi \text{ yrs}^{-1}.$$

Indeed this value of ω was chosen so that $\cos(\omega t)$ has a wavelength of 1 years.

Next, it is given that

$$\kappa = 0.002 \text{ cm}^2/\text{s}.$$

We use the fact that

$$1 \text{ yr} = 3.15 \times 10^7 \text{ s}, \quad 1 \text{ cm} = 10^{-2} \text{ m}.$$

So

$$\kappa = \frac{2 \cdot 10^{-3} \cdot 10^{-4} \text{ m}^2}{3.15 \times 10^{-7} \text{ yr}} = (2/3.15) \text{ m}^2 \text{ yr}^{-1} \approx 0.635 \text{ m}^2 \text{ yr}^{-1}.$$

When distributing the typical mesh in the x direction, we want to make sure we go deep enough. We had found that putting the cellar in about 4m deep was enough, so as long as your code solve for a range of x values, say, between $x = 0$ and $x = 10$, this should be sufficient.

The script is given in the solutions folder.

Q4. von Neumann analysis

This note should be filled in soon. The analysis is similar to the analysis for the heat equation as done in the problem class.

Q5. EBMs and variable sun output

- (a) Regarding the resultant variation on the Earth's mean surface.

Solution

The variation seems to be very small. Ignoring units, we have

$$T = \left(\frac{1-a}{\sigma\gamma} \right)^{1/4} Q^{1/4} \approx 67.3505 \times Q^{1/4}.$$

Substituting the values of Q given this gives a temperature that ranges from $T = 289.499 \text{ K}$ to 289.580 K . Converting this to Celsius gives 16.35°C to 16.43°C

i.e. a fraction of a degree.

- (b) Regarding the variation using the Budyko model.

Solution

We obtain the range of approximately $T = 17.06^\circ\text{C}$ to 17.19°C .

- (c) Regarding why the disparity between actual surface measurements.

Solution

There are many ways to respond, but one major factor is the oceans, which are ignored in our model. The oceans provide a massive energy sink for the planet, and so we would expect that the minor variations of Q due to the 11-year cycle would be smoothed out by the effects of the oceans.

Q6. Phase line analysis

- a. Sketch the solution $T(t)$ of this equation for $t > 0$ if $T(0) = 230, 240, 260, 270$ and 300 .

Solution

You should be able to do this question by hand, but the following graph is generated via the accompanying Jupyter script in the solutions folder. The point is that once the steady-state solutions are known (dashed) then each solution given by the different initial condition can be approximated by whether it tends towards or away from the nearest fixed point.

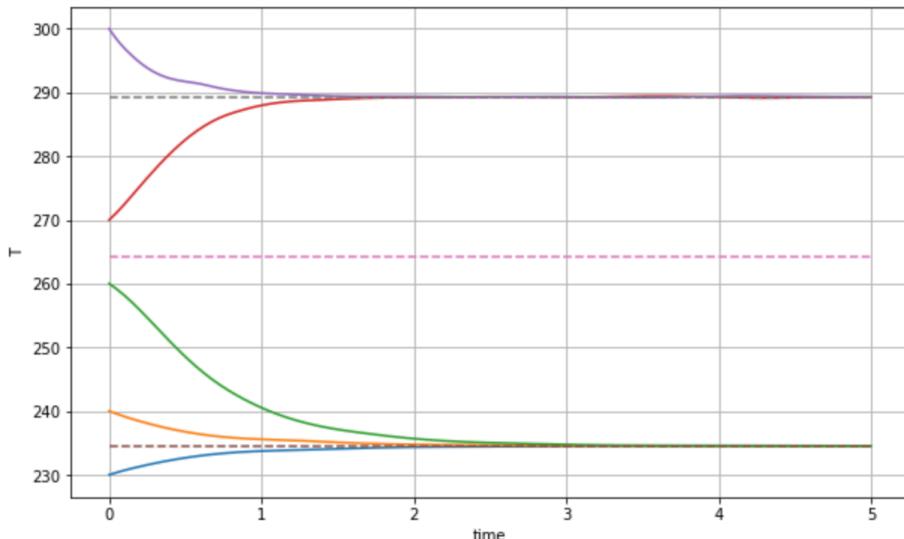


Figure 35.1: Graphical solution for Q3a

- b. Sketch the solution $T(t)$ of this equation for $t > 0$ if $T(0) = 285$. Then sketch the solution of this equation with the same initial data in the same coordinate system if C is twice as large. Explain your answer.

Solution

The below numerical solution shows blue for $C = 1$ and orange for $C = 4$. Notice that increasing C seems to decrease the rate of change of the evolution. Indeed, since

$$\frac{\partial T}{\partial t} \propto \frac{1}{C},$$

then multiplying C by factor is equivalent to slowing down the evolution by the inverse of that factor. A factor of C that is twice as large would slow down the evolution by half. Note that the oscillations shown in the graph below are numerical artifacts due to the tolerances on the ODE solver (how do you know this?)

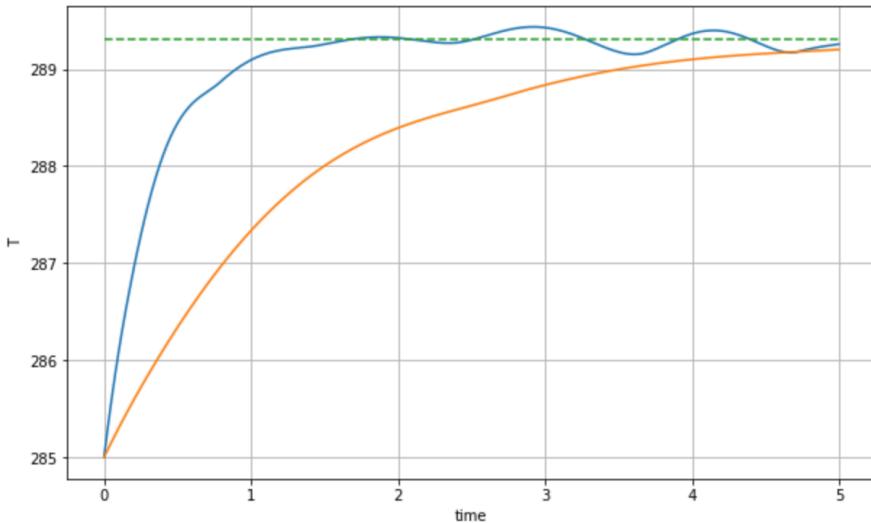


Figure 35.2: Graphical solution for Q3b

- c. If γ is decreased due to the increased greenhouse effect, the entire curve is shifted upwards. Sketch the solution if $T(0) = 280$. Sketch the solution with the same initial data if γ is decreased. Explain your answer.

Solution

Decreasing γ has the effect of shifting the curve upwards. The hottest steady state T_3 is consequently increased. There are two main effects. First, the steady-state is much hotter, so the system will tend towards a much hotter state. Second, because decreasing γ will increase the rate of change of T (since we are subtracting less via the factor $-\gamma T^4$), then the evolution towards the hotter state is initially much faster.

You see that in the following diagram.

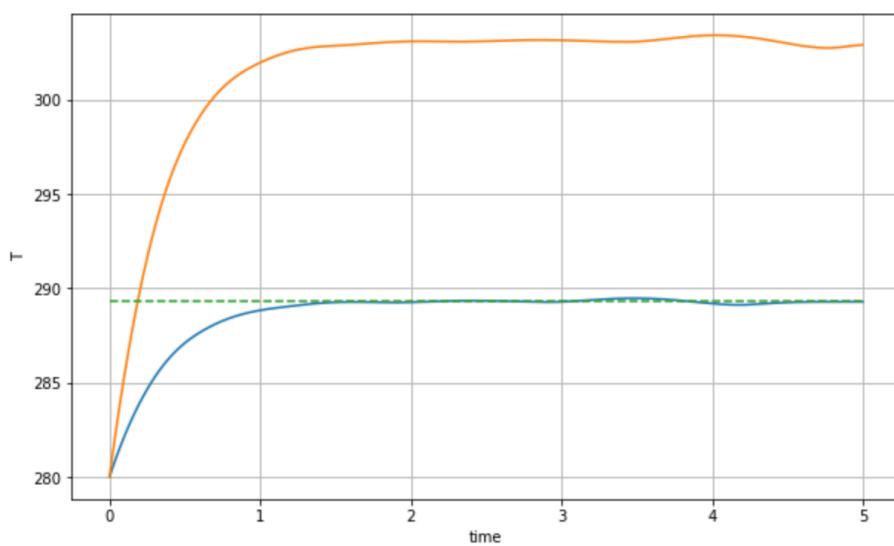


Figure 35.3: Graphical solution for Q3c

The above makes sense on a physical level. Decreasing γ is equivalent to increasing greenhouse gases. We do expect the system to tend to a hotter state, then.

CHAPTER 36

Problem set 5 solutions

Q1. Evolution

Consider again the basic time-dependent EBM given in (Equation 28.1). Let T^* be a steady-state solution and set $T = T^* + u(t)$ where $u(t)$ is a small perturbation from the steady state.

- Show that the perturbation satisfies

$$Cu = -Du + O(u^2).$$

and hence solve for the general solution of the leading-order perturbation (ignoring quadratic terms). What are the conditions on T^* so that the steady state is linearly stable?

Solution

We linearise via $T = T^* + u(t)$ where $u \ll 1$. Then we have

$$C \frac{\partial T}{\partial t} = G(T^*) + G'(T^*)u(t) + O(u^2).$$

The first term on the RHS is zero since T^* is a steady state. Ignoring quadratic terms, we see then that

$$Cu = -Du$$

where $D = -G'(T^*)$. The solution is thus

$$u(t) = Ae^{-Dt/C},$$

where A is a constant. In order to be linearly stable, we then need the above to tend to zero as $t \rightarrow \infty$, which is guaranteed if $D/C > 0$. Since $C > 0$ then we only need

$$G(T^*) < 0.$$

Figure 36.1: Graphical solution

- Assuming T^* is linearly stable, find the typical response time to a perturbation. For instance, what is the time it takes for the perturbation to reach the value $u(t) = 0$ if $u(0) = 1$? How does this response time change with C ? What is the physical interpretation of this regarding the climate?

Solution

The question is somewhat ill-posed. By the behaviour of the system, we know that it would take infinite time for the perturbation, u , to reach $u = 0$. However, we can still provide a sensible answer.

If the system starts from $u(0) = 1$, then to reach $u = \delta$, it would need to take the time

$$e^{-Dt/C} = \delta \implies t = -\frac{C}{D} \log \delta = \frac{C}{D} \log(1/\delta).$$

For instance, to reach $\delta = 1/2$ (half its original perturbation) would take $\log 2 \approx 0.69$ times (C/D).

This response is proportional to C so for instance, doubling C takes twice as long as before to reach the same value of δ . One can think of this as equivalent to decreasing the responsiveness of the planet to potential changes of stability.

Figure 36.2: Graphical solution

Q2. Integral of energy over the planet

Ignoring the effects of albedo, the total radiation absorbed over the surface of the planet (per unit time) is given by

$$\iint_{\text{planet}} Q s(y = \sin \varphi) dS.$$

This is what is known as a surface integral (Section 47.1). In the case of the spherical coordinate system, this is calculated by

$$\int_{\theta=0}^{2\pi} \int_{\varphi=-\pi/2}^{\pi/2} Q s(y = \sin \varphi) R_E^2 \cos \varphi d\varphi d\theta.$$

Use the properties of $s(y)$ in (Equation 14.3) to conclude that the total radiation absorbed is $4\pi R_E^2 Q$.

Solution

This is a straightforward calculation without tricks. Since Q is constant, we have

$$I = 2\pi R_E^2 Q \int_{\varphi=-\pi/2}^{\pi/2} s(y = \sin \varphi) \cos \varphi d\varphi.$$

Converting now to integration in y , we have

$$I = 2\pi R_E^2 Q \int_{y=-1}^1 s(y) dy.$$

Examining the function $s(y)$ in [Equation 14.3](#) we note that it is symmetric in y and has been defined so its integral is one. Thus

$$I = 4\pi R_E^2 Q.$$

This is the solar constant multiplied by the surface area of the planet (a sphere).

Figure 36.3: Graphical solution

Q3. Mean temperature in the latitude-dependent EBM

Consider now the latitude-dependent EBM

$$C \frac{\frac{\partial T}{\partial t}}{\frac{\partial t}{\partial t}} = Q s(y)[1 - a(y)] - (A + BT) + k(\bar{T} - T).$$

Recall the albedo is given by $a = a_i$ for $y > y_s$ and $a = a_w$ for $y < y_s$.

- a. By integrating the above equation over $y \in [0, 1]$, show that the mean temperature is given by

$$C \frac{d\bar{T}}{dt} = Q(1 - \bar{a}) - (A + B\bar{T}), \quad (36.1)$$

where

$$\bar{a} = \int_0^1 s(y)a(y) dy = \alpha_w \int_0^{y_s} s(y) dy + a_i \int_{y_s}^1 s(y) dy.$$

Solution

We have from integration,

$$C \frac{\partial}{\partial t} \int_0^1 T dy = Q \int_0^1 s(y) dy - Q \int_0^1 s(y)a(y) dy - (A + B\bar{T}) + k\bar{T} - k\bar{T}.$$

Simplifying and using the property of s gives

$$C \frac{\partial \bar{T}}{\partial t} = Q(1 - \bar{a}) - (A + B\bar{T}),$$

where we have defined

$$\bar{a} = \int_0^1 s(y)a(y) dy = \int_0^{y_s} s(y)a_w dy + \int_{y_s}^1 s(y)a_i dy.$$

Figure 36.4: Graphical solution

- b. In the case that s is given by (Equation 14.3), show that

$$\bar{a} = a_i + (a_w - a_i)y_s[1 - 0.241(y_s^2 - 1)]. \quad (36.2)$$

What is \bar{a} in the two situations of a completely ice-covered world and an ice-free world?

Solution

Again this question is free of tricks and only involves the integration using

$$s(y) = 1 - S_2 P_2(y)$$

where $S_2 = 0.482$ and $P_2(y) = \frac{1}{2}(3y^2 - 1)$. It is useful to compute the two quantities

$$\int_0^{y_s} s(y) dy = y_s - \frac{S_2}{2}(y_s^3 - y_s), \quad (3.1)$$

$$\int_{y_s}^1 s(y) dy = 1 - y_s + \frac{S_2}{2}(y_s^3 - y_s). \quad (3.2)$$

Multiply the first quantity by a_w and add to the second multiplied by a_i and simply to obtain

$$\bar{a} = a_i + (a_w - a_i)y_s \left[1 - \frac{S_2}{2}(y_s^2 - 1) \right].$$

Figure 36.5: Graphical solution

Q4. Sensitivity of the climate

Consider the equation for the global average temperature given in (Equation 36.1):

$$C \frac{d\bar{T}}{dt} = Q(1 - \bar{a}) - (A + B\bar{T}).$$

We would like to understand how the climate behaves under a small perturbation in the solar forcing. In general, though, the ‘constants’, A , B , and \bar{a} will depend on T . For example, their values may have been calibrated under a static situation. Therefore, let $A = A(T)$, $B = B(T)$, and $\bar{a} = \bar{a}(T)$.

Below, we drop all bars for convenience.

- a. Consider a perturbation of the solar radiation, say $Q = Q_0 + \delta$ where δ is small in comparison to Q_0 . Expand now the temperature into a series:

$$T = T_0 + \delta T_1 + \dots$$

Show that at $O(\delta)$, the perturbation is governed by

$$C \frac{dT_1}{dt} = (1 - a(T_0)) - B(T_0)T_1 - A'(T_0)T_1 - T_0 B'(T_0)T_1 - Q_0 a'(T_0)T_1.$$

Solution

Each of the functions should be expanded in the usual way. For example, from Taylor series we have

$$A(T) = A(T_0) + A'(T_0)(T - T_0) + O(T - T_0)^2.$$

Then using $T = T_0 + \delta T_1 + \dots$, we have

$$A(T) = A(T_0) + \delta A'(T_0)T_1 + O(\delta)^2.$$

Do the same for both $a(T)$ and $B(T)$. Then substituting into the equation gives (remember to drop bars):

$$\text{LHS} = \delta C \frac{dT_1}{dt} + O(\delta^2)$$

while for the right hand-side

$$\text{RHS} = Q_0(1 - a(T_0)) - (A(T_0) + B(T_0)T_0) \quad (36.3)$$

$$+ \delta [(1 - a(T_0)) - Q_0 a'(T_0)T_1 - A'(T_0)T_1 - B'(T_0)T_0 T_1 - B(T_0)T_1] + O(\delta)^2. \quad (36.4)$$

Now the first, $O(1)$, grouping of terms above is zero by assumption that T_0 is steady-state. So considering only those terms that are multiplied by δ , we have

$$C \frac{dT_1}{dt} = (1 - a(T_0)) - B(T_0)T_1 - A'(T_0)T_1 - T_0 B'(T_0)T_1 - Q_0 a'(T_0)T_1.$$

b. Consequently, show that the temperature perturbation can be written as

$$B(T_0)\tau \frac{\delta T_1}{\delta t} = [1 - a(T_0)] - \frac{B(T_0)}{g} T_1, \quad (36.5)$$

where

$$\begin{aligned}\tau &= \frac{C}{B(T_0)} \\ g &= \frac{1}{1-f} \\ f &= f_1 + f_2 \\ f_1 &= \frac{1}{B(T_0)} (-T_0 B'(T_0) - A'(T_0)), \\ f_2 &= \frac{1}{B(T_0)} (-Q_0 a'(T_0))\end{aligned}$$

The parameter τ measures the time scale of the climate system's thermal inertia. It involves the thermal inertia of the atmosphere and also the much larger inertia of the oceans. The factor g is the climate gain; it amplifies any response to the radiative perturbation by a factor of g . Also f_1 incorporates the effect of water-vapor feedback and f_2 that of ice and snow albedo feedback.

Solution

There is nothing tricky about this, but it just requires keeping track of bookwork. On the right hand-side, separate those terms proportional to T_1 and write it in the desired form. Refer to the lecture videos where we completed this (modulo a possible sign!)

c. Consider (Equation 36.5) at steady state, so therefore the perturbed equilibrium temperature is equal to

$$\delta T_1 = \frac{[1 - a(T_0)]\delta g}{B(T_0)}.$$

If the CO₂ level in the atmosphere doubles, then the radiative forcing might be adjusted as:

$$(1 - a(T_0))\delta = 3.7 \text{ W} \cdot \text{m}^{-2}.$$

Assuming that the climate gain is $g = 3$ and $B(T_0) = 1.9 \text{ W} \cdot \text{m}^{-2} \cdot (\text{ }^\circ\text{C})^{-1}$, what is the expected increase in temperature?

Solution

So we know that $(1 - a)\delta = 3.7 \text{ W/m}^2$. So according to the equation, we have the fact that

$$\delta T_1 = 3.7 \frac{3}{1.9} \approx 5.8 \text{ }^\circ\text{C}.$$

So this is the expected increase in temperature. Note that if a climate gain factor of $g = 1$ is used then this gives an expected increase of 1.9 degrees instead.. The range of current model predictions, in fact, is about 1.5 to 4.5 degrees so this is not bad.

CHAPTER 37

Problem set 6 solutions

Q1. A van der Pol equation

- a. Determine $f(x)$ so that this equation can be written as a Liénard phase plane system in the form

$$\begin{aligned}\epsilon \dot{x} &= f(x) + 4y, \\ \dot{y} &= -x.\end{aligned}$$

Solution

Start from the ODE and re-write

$$\frac{d}{dt} [\epsilon \dot{x} + (x^3 - 3x^2 - 9x)] = -4x.$$

Double check that the derivative of the LHS returns the original problem. Now set the quantity in brackets equal to one unknown:

$$\tilde{y} = \epsilon \dot{x} + (x^3 - 3x^2 - 9x),$$

and solve for the derivative:

$$\epsilon \dot{x} = \tilde{y} - (x^3 - 3x^2 - 9x).$$

The remaining equation is

$$\dot{\tilde{y}} = -4x.$$

Now in this question, we have chosen a slightly different choice to eliminate the factor of 4. Set $\tilde{y} = 4y$ to get the system

$$\epsilon \dot{x} = f(x) + 4y, \tag{37.1}$$

$$\dot{y} = -x. \tag{37.2}$$

It doesn't really matter which choice you select. So here we see $f(x) = -x^3 + 9x + 3x^2$. What we often call the slow manifold in lectures is then

$$y = -f(x)/4 = \frac{x^3 - 3x^2 - 9x}{4} \equiv S(x).$$

- b. For fixed $\epsilon > 0$, find the equilibrium point(s) in the phase plane, find their eigenvalues, and classify their linear stability.

Solution

The only equilibrium point is $(x_*, y_*) = (0, 0)$. Because the equilibrium point is at the origin, we can obtain the linear matrix by ignoring the higher-order powers in the differential equations. The matrix is

$$A = \begin{pmatrix} \frac{9}{\epsilon} & \frac{4}{\epsilon} \\ -1 & 0 \end{pmatrix}.$$

Subtract λ from the diagonal elements and solve the resultant quadratic equation $\lambda(\lambda - 9/\epsilon) + 4/\epsilon = 0$ to see that the eigenvalues are $\lambda = (9 \pm \sqrt{81 - 16\epsilon})/(2\epsilon) > 0$, so the point is an unstable node.

- c. Use the expansions $x(t) = x_0(t) + \epsilon x_1(t) + O(\epsilon^2)$ to determine the equations for the leading-order slow solution. Sketch the slow manifold, indicate the directions of motion on each part, and identify the two attracting points on the curve.

Solution

As usual expand the solutions into a series in powers of ϵ . The slow manifold is $y_0(x_0) = S(x_0) = (x_0^3 - 3x_0^2 - 9x_0)/4$. In image of the slow manifold is shown below in blue. Notice that for $x < 0$, y is increasing and for $x > 0$, y is decreasing (due to the second differential equation).

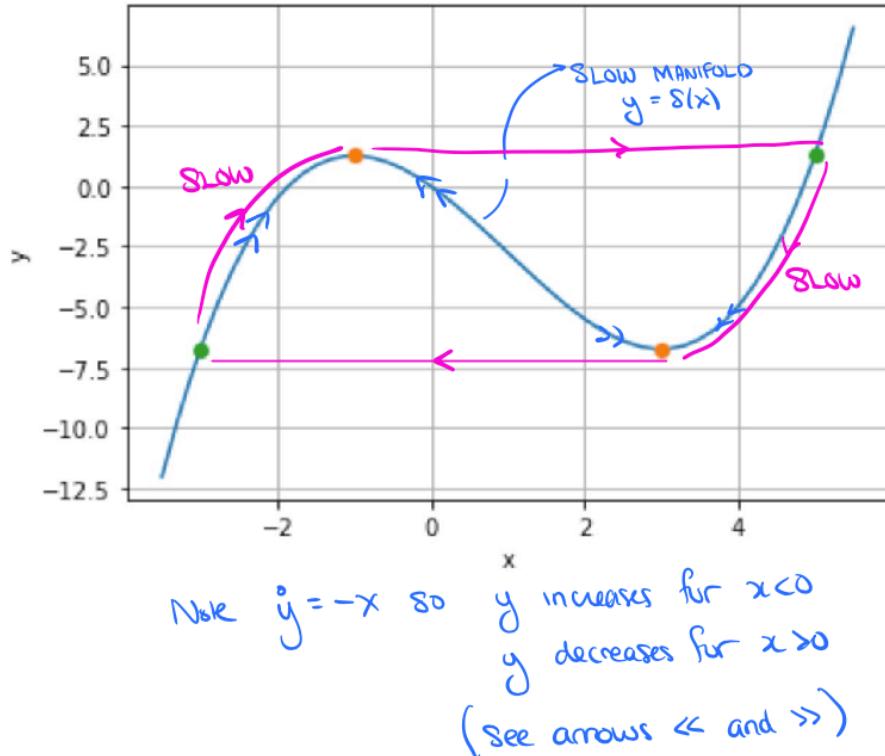


Figure 37.1: Illustration of slow manifold

- d. Use the expansions $x(t) = X_0(T) + \epsilon X_1(T) + O(\epsilon^2)$ and $y(t) = Y_0(T) + \epsilon Y_1(T) + O(\epsilon^2)$ with $T = t/\epsilon$ to obtain the leading-order fast solution.

Solution

Rescale $t = \epsilon T$ and re-write $x = X(T)$ and $y = Y(T)$. We get

$$\frac{dX}{dT} = f(X) + 4Y \quad (37.3)$$

$$\frac{dY}{dT} = -\epsilon X. \quad (37.4)$$

Now expand into powers of ϵ and take the leading-order:

$$\frac{dX_0}{dT} = f(X_0) + 4Y_0 \quad (37.5)$$

$$\frac{dY_0}{dT} = 0. \quad (37.6)$$

The second equation indicates that Y_0 is constant. Its value, found in part (e) is either $5/4$ or $27/4$. The remaining equation for X_0 is nonlinear and there is not much you can do with it. The important item of understanding is to examine its form:

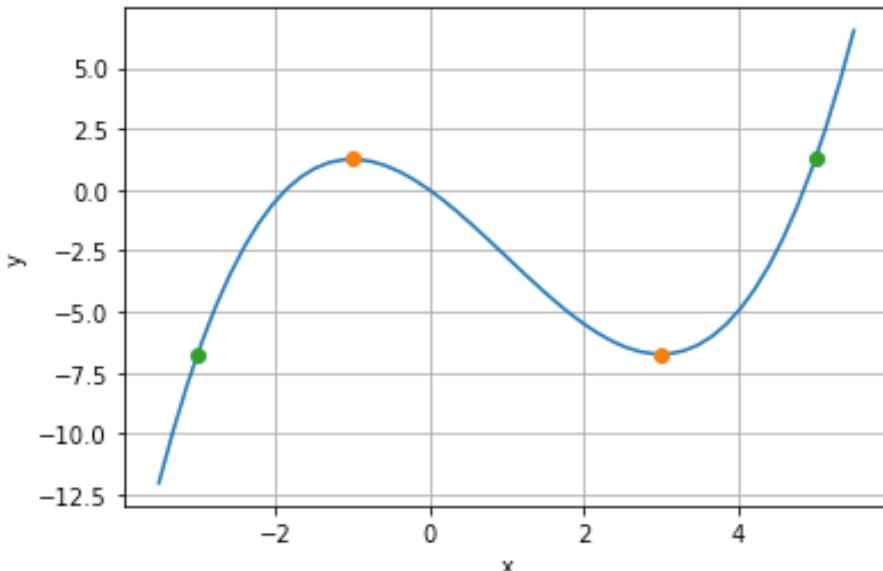
$$\frac{dX_0}{dT} = f(X_0) + 4Y_0.$$

Now on the upper branch of the picture shown above $f(X_0) = 4Y_0 > 0$ and therefore the solution increases its value in X_0 until it intersects with the blue slow manifold, at which point its rate of change is zero and it descends the slow manifold. A similar behaviour occurs on the lower branch of the fast evolution.

- e. Use the phase plane to determine the maximum and minimum values of $x(t)$ during an oscillation. Sketch $x(t)$ as a function of time.

Solution

By checking the derivatives, we can see that the slow manifold has local extrema at $(-1, 5/4)$ and $(3, 27/4)$. In the fast evolution stages, y is constant and x evolves from one extrema value to the other on the slow manifold, satisfying $y = S(x)$. Here is a plot of the key points.



Q2. Fast-slow dynamics with three variables

a. For the system

$$\begin{aligned}\dot{x} &= 2 - y, \\ \dot{y} &= x - z, \\ \epsilon \dot{z} &= y - y^2 + \frac{1}{3}y^3 - z,\end{aligned}$$

with the initial conditions of $x(0) = 1$, $y(0) = 3$, and $z(0) = 0$.

Identify the surface $z = S(x, y)$ that defines the slow manifold. Find the equilibrium point of the leading-order slow phase plane system and show that it is asymptotically stable for $t \rightarrow \infty$. Also determine the form of the initial layer that describes the transition from the initial conditions to the slow manifold.

Solution

Setting $\epsilon = 0$ we see that the slow manifold is $z = S(y) = y - y^2 + y^3/3$. The leading-order slow system is

$$\frac{dx_0}{dt} = 2 - y_0 \tag{37.7}$$

$$\frac{dy_0}{dt} = x_0 - y_0 + y_0^2 - y_0^3/3. \tag{37.8}$$

\$\$\text{The only equilibrium point is } (x^*, y^*) = (2/3, 2). \text{ Linearisation gives eigenvalues of } \lambda = 1/2(-1 \pm 3i) \text{ so the point is a stable spiral. Consequently the } t \rightarrow \infty \text{ solution will be } (x, y, z) = (2/3, 2, 2/3). \text{ Note also that } 2/3 = S(2).

Near the initial condition there is a boundary layer, found by setting

$$t = \epsilon T,$$

and transforming $x = X(T)$, $y = Y(T)$, $z = Z(T)$. We get at leading order,

$$\frac{dX_0}{dT} = 0 \quad (37.9)$$

$$\frac{dT_0}{dT} = 0 \quad (37.10)$$

$$\frac{dZ_0}{dT} = Y_0 - Y_0^2 + Y_0^3/3 - Z_0. \quad (37.11)$$

The initial condition sets $X_0 = 1$ and $Y_0 = 3$. We can finally solve the equation for Z_0 to get

$$Z_0 = 3(1 - e^{-T}).$$

b. For the system

$$\dot{x} = 2 - y,$$

$$\epsilon\dot{y} = x - z,$$

$$\dot{z} = y - y^2 + \frac{1}{3}y^3 - z,$$

with the initial conditions of $x(0) = 0$, $y(0) = 3$, and $z(0) = 1$.

Show that the slow manifold reduces to a curve that could be written in parametric form as $x = x(z)$, $y = y(z)$, $z = z$. Determine the asymptotic solution for $t \rightarrow \infty$. Also determine the form of the initial layer that describes the transition from initial conditions to the slow manifold.

Solution

Setting $\epsilon = 0$, we see that the second equation gives $x = z$. Now if you use this result in the first equation you obtain $\dot{z} = 2 - y$. This is now equated with the third equation which yields an algebraic equation:

$$2 - y = y - y^2 + \frac{y^3}{3} - z \implies z = \frac{y^3}{3} - y^2 + 2y - 2. \quad (37.12)$$

Therefore we conclude that $z = z(y)$. We can verify that this equation is monotonically increasing in y , so we can invert without issue, yielding $y = y(z)$. Similarly, we obtain $x = z$. So the slow manifold is given by

$$(x, y, z) = (z, y(z), z),$$

which is a curve in 3D.

In order to determine what happens as $t \rightarrow \infty$, differentiate

$$\frac{dy}{dt} = \frac{dy}{dz} \frac{dz}{dt} = \frac{2 - y}{2 - 2y + y^2},$$

which follows from using the first equation of the system of ODEs and also the z derivative of (Equation 37.12). In particular, note that we do not have to invert this latter equation and we can take the y -derivative of (eq-PS5-Q3-tmp) and use

$$\frac{1}{\frac{dz}{dy}} = \frac{1}{2 - 2y + y^2}.$$

Take $dy/dt = 0$ for the steady state gives

$$y^* = 2,$$

as the only fixed point. At this value of y , notice that $z^* = 2/3$ from (Equation 37.12). So the fixed point is

$$(x^*, y^*, z^*) = (2/3, 2, 2/3).$$

The procedure for the initial layer is the same as in part (a). This time, the fast system satisfies at leading order,

$$\frac{dX_0}{dT} = 0 \quad (37.13)$$

$$\frac{dY_0}{dT} = X - Z \quad (37.14)$$

$$\frac{dZ_0}{dT} = 0. \quad (37.15)$$

So we obtain $X_0 = 0$ and $Z_0 = 1$ using the initial condition. Solving for the final equation for Y_0 with $Y_0(0) = 3$ gives

$$Y_0 = -T + 3.$$

CHAPTER 38

Problem set 7 solutions

Q1. Non-dimensionalisation of the Stommel box model

Consider the Stommel box model given in (Equation 23.1). Non-dimensionalise the model in order to produce the set of equations given in (Equation 23.2), repeated here:

$$\begin{aligned}\frac{dx}{dt} &= \delta(1 - x) - |f(x, y)|x, \\ \frac{dy}{dt} &= 1 - y - |f(x, y)|y,\end{aligned}$$

where we have introduced the function,

$$f(x, y; R, \lambda) = \frac{1}{\lambda}(Rx - y),$$

Provide a brief description of the physical interpretations of the parameters λ , δ , and R .

Solution

The non-dimensionalisation procedure was done in the lecture notes and lectures. The final parameters were established by

$$\delta = \frac{d}{c}, \quad \lambda = \frac{c}{2\alpha k \Delta T^*}, \quad R = \frac{\beta \Delta S^*}{\alpha \Delta T^*}.$$

Their interpretations are as follows: δ measures the relative relaxation temporal rates of the salt-basin exchange and the temperature-basin exchange. Since thermal energy turns out to exchange on a faster time scale than salinity, then we are typically interested in $\delta \ll 1$.

The parameter λ is a measure of the strength of the THC. For example, with k large, this corresponds to significant flow through the pipes connecting the boxes, and so $\lambda \rightarrow 0$ corresponds to strong THC.

Finally R allows us to compare the relative effects of temperature and salinity differences between the external basins. Salinity differences dominate if $R > 1$ and temperature differences dominate if $R < 1$.

Q2. Investigations for R

Using the script available on Noteable investigate the behaviour of the Stommel box model under changing values of R and fixed values of the other parameters. Based on your intuition, what do you expect to be the features of the bifurcation diagram, as shown in the (R, f) -plane?

Solution

Remember that the steady states are found by examining the intersections of the curves $\lambda f = \phi$. From (Equation 23.5), we have

$$\phi(f^*) = \frac{\delta R}{\delta + |f^*|} - \frac{1}{1 + |f^*|}.$$

so essentially the effect of changing R is to multiply the curve ϕ by a multiplier. Using the [code on the JupyterHub](#) we obtain the following graph for different values of R .

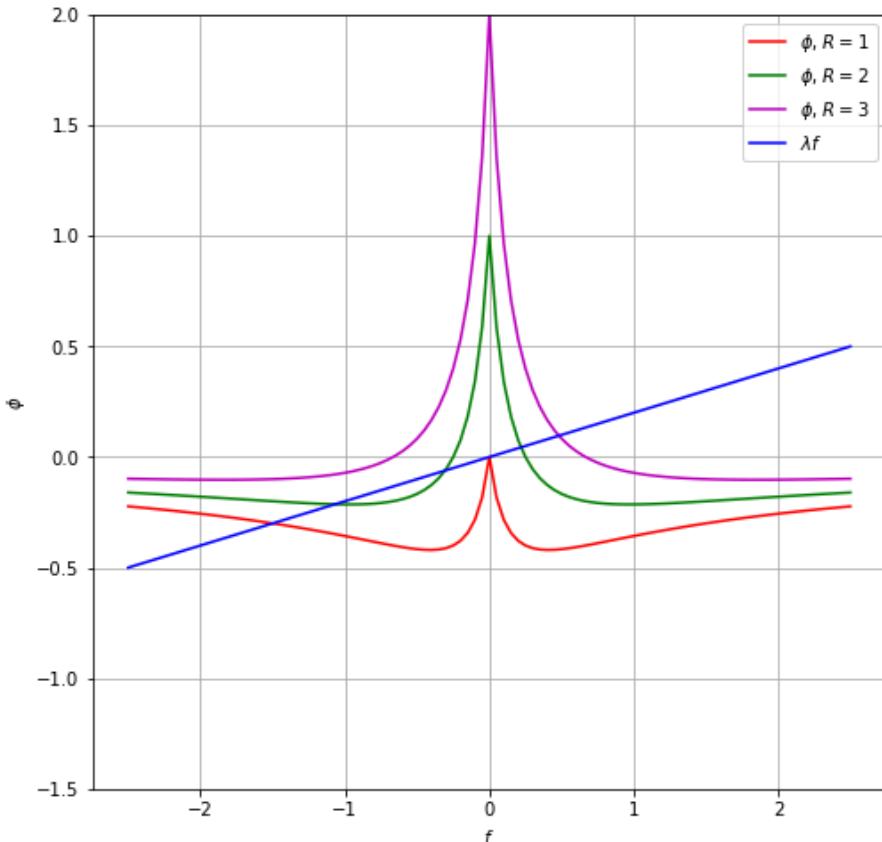


Figure 38.1: Steady state intersections for Stommel's box model

For the smallest values of R , note that only the lowermost steady state will be present. As R crosses the critical value of $R = 1$, two intersections are borne at $\phi = 0$. Eventually R crosses another critical value, and the two leftmost intersections merge then disappear, leaving only the largest steady-state.

Because the stability of the solution is essentially given by the ordering of the curves (if the curve λf is larger than the curve ϕ on the left of the steady-states), then we can see that, as typical for stability in this course, the first and third steady states are stable, and the middle one unstable.

We can now plot a bifurcation diagram using the above information.

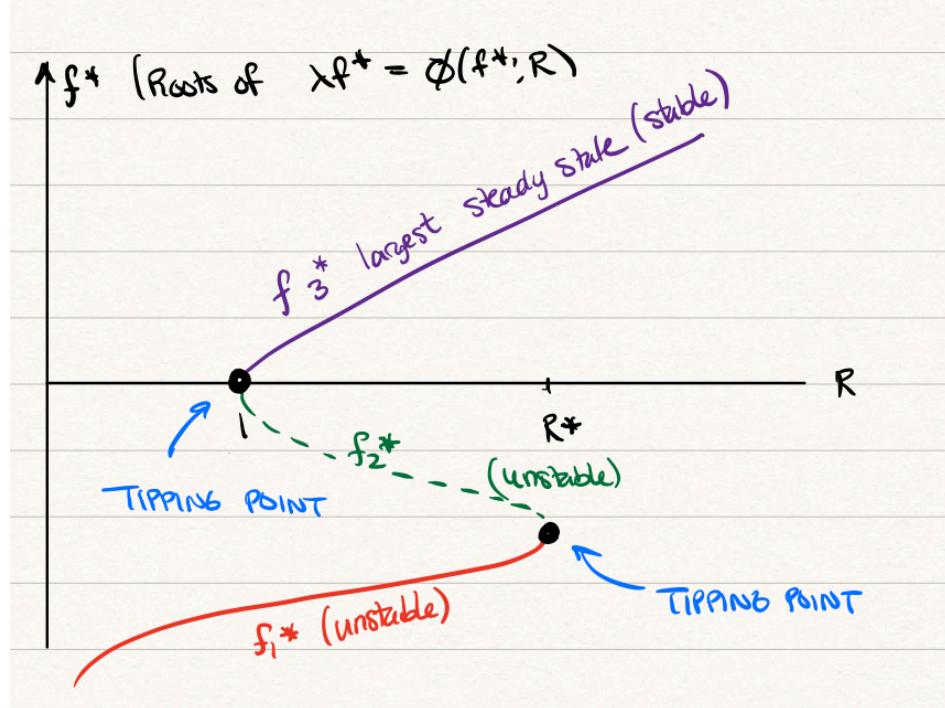


Figure 38.2: Bifurcation diagram for R vs f^*

Q3. Alternative scalings of the Stommel box model

It is possible to scale the problem differently, and this may allow for simpler analysis. Consider the set of equations in Q1. Write instead

$$x = [x]s, \quad y = [y]\theta, \quad t = [t]t',$$

Choose the correct scalings, $[x]$, $[y]$, and $[t]$, so that we obtain a re-scaled version:

$$\begin{aligned} \frac{ds}{dt'} &= 1 - (\epsilon + |q|)s, \\ \frac{d\theta}{dt'} &= 1 - (\mu + |q|)\theta, \end{aligned}$$

where now the flow term is given by

$$q = \kappa(-\theta + \tilde{R}s).$$

In addition to the choice of scalings, you will need to describe how the new parameters, ϵ , μ , κ , and \tilde{R} , are related to the older parameters.

Solution

Putting in the scalings,

$$\begin{aligned}\frac{[x]}{[t]} \frac{ds}{dt'} &= \delta(1 - [x]s) - [x]|f(x, y)|s, \\ \frac{[y]}{[t]} \frac{d\theta}{dt'} &= 1 - [y]\theta - |f(x, y)|\theta.\end{aligned}$$

Comparing the equations, we see that we should choose

$$[x] = \epsilon, \quad \frac{\delta[t]}{[x]} = 1, \quad [t] = \mu, \quad [y] = [t] = \mu.$$

Under this choice,

$$\begin{aligned}\frac{ds}{dt'} &= 1 - \epsilon s - [t]|f(x, y)|s, \\ \frac{d\theta}{dt'} &= 1 - \mu\theta - [t]|f(x, y)|\theta.\end{aligned}$$

The latter quantity is

$$\mu f = \frac{\mu[y]}{\lambda} \left(-\theta + \frac{R[x]}{[y]} s \right).$$

We there write

$$\kappa = \frac{\mu[y]}{\lambda}, \quad \tilde{R} = \frac{R[x]}{[y]}$$

Thus we have produced

$$\frac{ds}{dt'} = 1 - \epsilon s - |q|s, \tag{38.1}$$

$$\frac{d\theta}{dt'} = 1 - \mu\theta - |q|\theta. \tag{38.2}$$

where

$$q = \kappa(-\theta + \tilde{R}s).$$

Q4. Box models for flooding

Solution

Solution shown in lectures.

Part VII

Problem classes

Problem class 1: an introduction to Noteable

During this module, we will perform various computational experiments using Python. In order to make it easy for students to code in Python from a variety of devices (desktops, laptops, tablets, and even a phone), we use a cloud-computing system called Noteable that allows you to code from within a web browser. Students who like to code using their own Python setup on their computer can also do so. The goal of this problem class is to quickly get you up to speed with the basic Noteable system, and how to manipulate outputs. We should cover:

- A walkthrough of setting up with Noteable
- Basic plotting using matplotlib and numpy
- How to export graphics and worksheets

39.1 Getting started with Noteable

Use your [Moodle course page](#) to access the Noteable Python interface by clicking on the appropriate link in the course materials. Follow the instructions on the Moodle page to add the Git repository to your directory.

1. Navigate to the welcome screen for MA30287 and follow the instructions in `Welcome.ipynb` and `FirstTimeSetup.ipynb` in order to create your own local directory at `/MA30287_workspace/`
2. Navigate to the workspace folder. Click `New -> Notebook`. If asked, select the kernel `Python 3 (ipykernel)`.
3. In the first line of input, select, in toolbar `Code -> Markdown`. This allows you to annotate your notebook with Markdown-style text input.
In the first line of input, type `# Problem class 1`. Then either type `Shift + Enter` or press the play button which will execute the line(s) of input. Your markdown text should render as a nicely formatted entry.

4. Rename the file to something appropriate, like `problemclass01`. You can do this by right-clicking the filename in the file manager and selecting `Rename`.

Proceed to the next question.

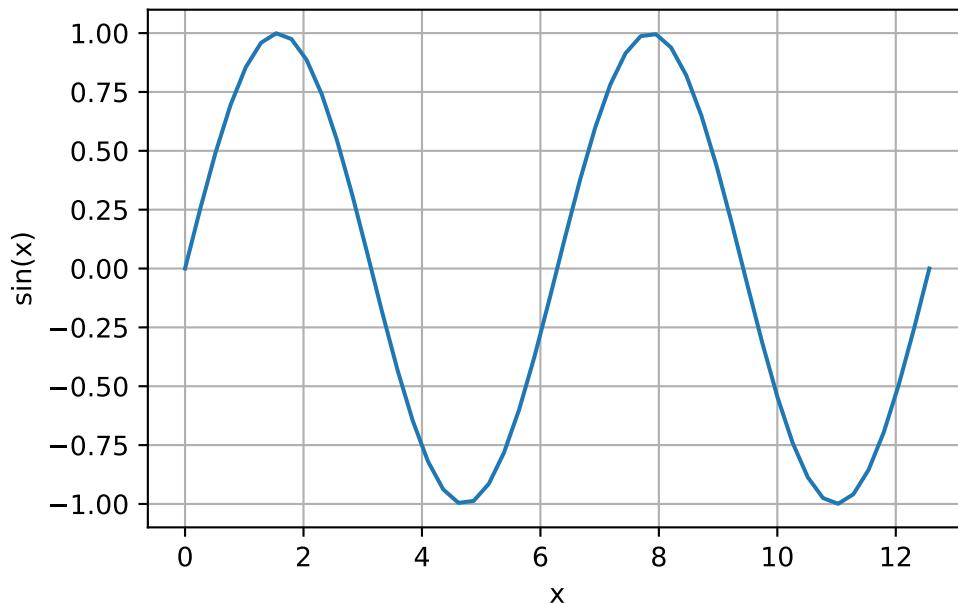
39.2 Mathematical plotting

Three key Python packages used throughout this course are `numpy`, `matplotlib`, and `scipy`. Numpy provides array functionality, allowing you to naturally manipulate vectors and matrices. Matplotlib is a general plotting package for Python, allowing for the creation of (typically) 2D and 3D plots. And `scipy` provides algorithms for differential and integral operations allowing e.g. the solution of differential equations.

Here is a script to generate a graph of a sine curve.

```
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 4*np.pi, 50)
y = np.sin(x)
plt.plot(x, y)
plt.xlabel('x');
plt.ylabel('sin(x)');
plt.grid(1)
```



Here is a script to generate a 3D plot.

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

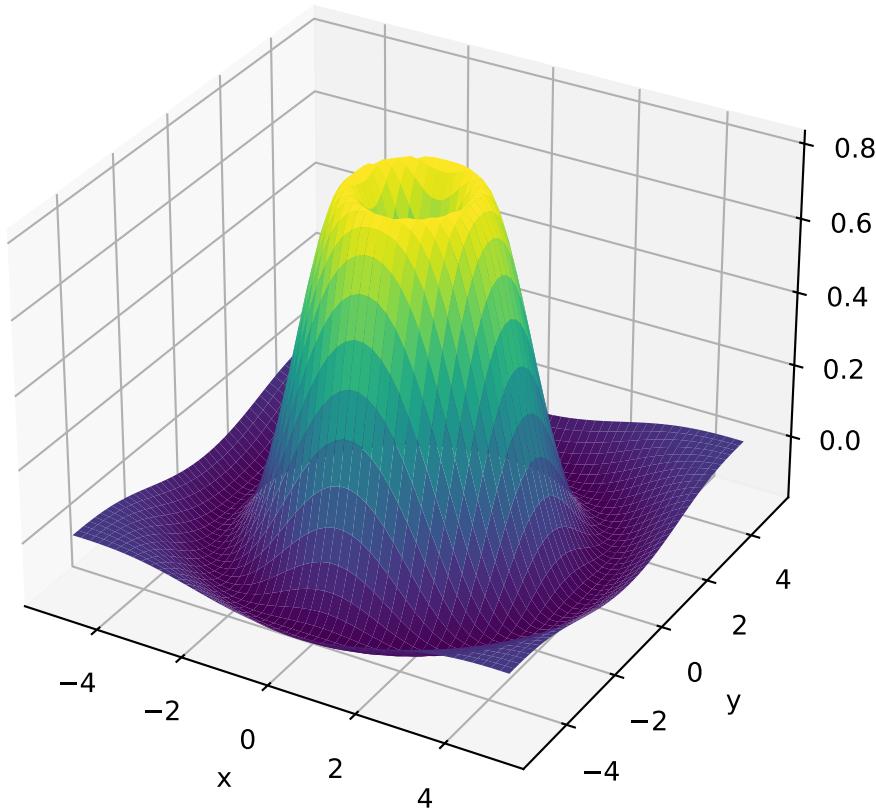
# Define the function
def f(x, y):
    return np.sin(np.sqrt(x**2 + y**2)) * np.exp(- (x**2 + y**2) / 10)

# Create grid
x = np.linspace(-5, 5, 50)
y = np.linspace(-5, 5, 50)
X, Y = np.meshgrid(x, y)
```

```
Z = f(X, Y)

# Plot the surface
fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(X, Y, Z, cmap='viridis')

# Labels and title
ax.set_xlabel("x")
ax.set_ylabel("y")
ax.set_zlabel("z")
plt.show()
```



39.3 Debugging Jupyter code

It would be good to demonstrate a workflow to understand how errors are diagnosed and studied (e.g. by printing simple variables throughout the script). Also, it would be good to show students how graphs can be zoomed in/out and/or rotated within the browser interface (this may be tricky in Noteable).

39.4 Outputting and saving

During the problem class, we will also show how to:

- Create a zip file of your workspace using the terminal so that you can save a copy of your files onto your local filesystem. This can be done by running the script ExportMA30287.ipynb in the root directory of the Noteable system, and then downloading the result.
- Create a copy of your script in PDF form so that it can be sent around as a single document. This can be done by going to File -> Save and Export Notebook As...

39.5 Challenge coding exercise

Create an animation of the 2D or 3D plot, e.g. by making it so the waves ‘ripple’. Using a `for` loop, animate the wave. Then output the animation as a video file.

CHAPTER 40

Problem class 2: dimensional analysis

Abstraction

It may seem strange to study examples from elementary physics in a course that is supposed to be about Planet Earth. But simple examples are the best ways to learn these important techniques. The full climate equations are often very involved. These toy models still nevertheless capture the spirit of what you must do when attacking any scientific problem.

In this problem class, we will practice some concepts about non-dimensionalising in preparation for the Problem Set 1 in Chapter 25. We will cover these two strategies in choosing scalings.

Scaling principle 1

Select the characteristic scales so that as many of the possible non-dimensional numbers, Π_i , $i = 1, 2, 3, \dots$ are normalised.

Scaling principle 2

Select characteristic scales so that no terms in the model diverge in the physical limit of interest.

Here are the problems we shall do in the problem class.

40.1 Projectile motion

A projectile of mass M (in kg) is launched vertically with initial velocity V_0 (in m s^{-1}) from a position Y_0 (in m) above the surface. Thus the mass's position, $Y(t)$ is governed by Newton's second law (applied to the mass and the mass of the Earth) and the set of equations

$$MY_{tt} = -\frac{gR_E^2 M}{(R_E + Y)^2},$$
$$Y(0) = Y_0,$$

where $g = 9.81 \text{ m s}^{-2}$ and $R_E = 6.4 \times 10^6 \text{ m}$ is the radius of the Earth.

1. Non-dimensionalise the equation using arbitrary length and time scales.
2. Identify the non-dimensional constants, Π_i .
3. Choose a length scale of $L = Y_0$ and time scale of $T = (L/g)^{1/2}$. Discuss the resultant equation and the interpretation of choosing these scales.
4. Does your above choice allow you to easily study the limit of $R_E \rightarrow \infty$? If the limit can be taken, reduce the governing system to a simpler equation.
5. Does your choice in Q3 allow you to easily study the limit of $Y_0 \rightarrow 0$? If not, choose an alternative choice of length and time scales and in that case, reduce the set of equations.

40.2 Terminal velocity

A ball of radius R (in m) and uniform density ρ (in kg m^{-3}) falls in a viscous fluid. The fluid has density ρ_f (in kg m^{-3}) and viscosity (a measure of friction or resistance) μ (in $\text{kg m}^{-1} \text{s}^{-1}$). The equation that governs the velocity is

$$\frac{4}{3}\pi R^3 \rho \frac{dV}{dt} = \frac{4}{3}\pi R^3 (\rho - \rho_f)g - 6\pi\mu RV,$$

$$V(0) = V_0.$$

1. Choose appropriate velocity and time scales to non-dimensionalise the equation so as to leave only a single non-dimensional number on the drag term (the last term on the right hand-side).
2. Define the non-dimensional parameter expressing a ratio between drag force and gravity force by the Stokes number (St) and confirm that it is

$$St = \frac{9\mu V_0}{2(\rho - \rho_f)gR^3}.$$

3. Comment on the two limits of $St \rightarrow 0$ and $St \rightarrow \infty$. Can the problem be reduced in these two limits? If so, reduce and solve.

40.3 Navier-Stokes Equations

Consider the Navier-Stokes equations (which is a set of equations that dictates fluid flow):

$$\frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\frac{1}{\rho} \nabla p + \nu \nabla^2 u.$$

The parameters and variables are as follows:

- $x = (x, y, z)^T$ is the direction field (each with dimension m)
- $u = (u, v, w)^T$ is the velocity field where u is the movement along the x -direction, v is the movement along the y -direction and w is the movement along the z -direction (each with dimension m s^{-1})
- t is time (dimension s)
- ρ is the fluid density (dimension kg m^{-3})
- ν is the dynamic viscosity, which represents the resistance to flow due to gravity (with dimension $\text{m}^2 \text{s}^{-1}$)

- p is the internal pressure of the fluid (dimension $\text{kg m}^{-1} \text{s}^{-2}$)
- $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)^T$ is the gradient operator (each with the with effective dimension m^{-1}).
- $\frac{\partial}{\partial t}$ is the partial time derivative (with effective dimension s^{-1})
- $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ is the Laplacian operator (with effective dimension m^{-2}).

1. Non-dimensionalise the equation using arbitrary length, time, speed and pressure scales.

Non-dimensionalisation

Consider the following arbitrary non-dimensionalisation:

$$x = [x]x', \quad u = [u]u', \quad t = [t]t', \quad p = [p]p'$$

where $[x'] = [u'] = [t'] = [p'] = 1$ are the new non-dimensional quantities whereas $[x] = \text{m}$, $[u] = \text{m s}^{-1}$, $[t] = \text{s}$ and $[p] = \text{kg m}^{-3}$.

Replacing these into the Navier-Stokes equation gives

$$\frac{[u]}{[t]} \frac{\partial u'}{\partial t'} + \frac{[u]^2}{[x]} (u' \cdot \nabla') u' = -\frac{[p]}{[x]\rho} \nabla' p' + \frac{\nu[u]}{[x]^2} \nabla'^2 u'$$

where ∇' is the gradient operator in the new coordinate space (x', y', z') , i.e. $\nabla' = \left(\frac{\partial}{\partial x'}, \frac{\partial}{\partial y'}, \frac{\partial}{\partial z'} \right)^T$. Similarly, the operator ∇'^2 is the Laplacian in the new coordinate space, i.e. $\nabla'^2 = \frac{\partial^2}{\partial x'^2} + \frac{\partial^2}{\partial y'^2} + \frac{\partial^2}{\partial z'^2}$.

2. Rearrange the equation so that the coefficient of the time derivative is equal to 1.

Time Derivative Coefficient

The dashes will be dropped henceforth for ease of notation. Rearranging to make the coefficient of the time derivative 1 gives

$$\frac{\partial u}{\partial t} + \frac{[u][t]}{[x]} (u \cdot \nabla) u = -\frac{[p][t]}{[u][x]\rho} \nabla p + \frac{\nu[t]}{[x]^2} \nabla^2 u.$$

3. Choose an appropriate time scale so that the coefficient of the pressure derivative is 1, then rearrange until only one term in the equation has a coefficient.

Pressure Derivative

In order to eliminate the coefficient of the pressure derivative, suppose that the choice of the non-dimensionalisation is

$$\frac{[p][t]}{\rho[x][u]} = 1.$$

This means that the term $[p]$ can be written as

$$P = \frac{\rho[u][x]}{[t]}.$$

Checking the dimensions:

$$[P] = [\rho][u][x][t]^{-1} = (\text{kg m}^{-3}) (\text{m s}^{-1}) (\text{m}) (\text{s})^{-1} = \text{kg m}^{-1} \text{s}^{-2}$$

which is indeed the dimension for pressure.

Suppose that the term $\frac{[u][t]}{[x]}$ should also be equal to 1, then this would mean that the time non-dimensionalisation can be done by having $[t] = \frac{[x]}{[u]}$.

Checking the dimensions:

$$[t] = [x][u]^{-1} = (\text{m}) (\text{m s}^{-1})^{-1} = \text{s}$$

which is indeed the dimension for time.

Therefore, with these choices of $[t]$ and $[p]$, the Navier-Stokes equation becomes

$$\frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\nabla p + \frac{\nu}{[u][x]} \nabla^2 u.$$

4. Let $\text{Re} = \frac{[x][u]}{\nu}$, known as the **Reynolds number**. Show that Re is dimensionless and hence, write the final form of the differential equation using the Reynolds number.

Non-dimensionality of the Reynolds Number

$$[\text{Re}] = \frac{[x][u]}{[\nu]} = (\text{m}) (\text{m s}^{-1}) (\text{m}^{-2} \text{s}^{-1})^{-1} = 1$$

Therefore, this gives the non-dimensional Navier-Stokes equation for fluid flow

$$\frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\nabla p + \frac{1}{\text{Re}} \nabla^2 u.$$

5. Give an interpretation of the Reynolds number and what happens as $\text{Re} \rightarrow \infty$ and $\text{Re} \rightarrow 0$.

The Reynolds Number

The Reynolds number Re gives a measure for the effect of viscosity for fluid flow or for the movement through a fluid. As Re gets larger, the fluid becomes less viscous hence reducing the effect of viscosity on the movement and the inertial forces dominate. (The case when $\text{Re} \rightarrow \infty$ is called inviscid flow where there is no viscosity.) As Re decreases, the viscosity increases, making the movement depend on the viscous forces more than the inertial forces.

The Reynolds number is important in modelling since when length and speed scales change from full-sized objects to smaller models (like models of cars and buildings for testing purposes), the same environment cannot be translated from one to the other (since a full sized building in 20mph winds will not move whereas 20mph winds on a small scale of model of the building will be affected much more). However, if the Reynolds number is fixed in both cases, the model can be made

more realistic.

CHAPTER 41

Problem class 3: BVPs

This problem class will involve the completion of the ODE studied in asymptotic approximations III. We will then study the boundary-value problem which forms part of the third problem set, namely:

$$\epsilon T'' + 2T' + T = 0, \\ T(0) = 0 \quad \text{and} \quad T(1) = 1.$$

During the problem class, you will study both the solution of the above problem using stock routines in Python, and also in coding your own routine. The analysis of the above problem using asymptotics was done in Lecture 8.

CHAPTER 42

Problem class 4: PDEs and numerical stability

During the introduction to numerical methods on PDEs, it was mentioned that the first-order explicit Euler scheme is not well-suited for certain problems, such as the heat equation. In this chapter, we provide a brief note on deriving the famous CFL stability condition, which informs us of the timestep size required for a stable solution.

The following analysis is known as von Neumann analysis; it was originally derived in 1947 by Crank and Nicolson, and then later made rigorous by Charney *et al.* in 1950 (of which the third author was John von Neumann). You can read about the procedure [here](#).

Consider the finite-difference formulation of the heat equation, as applied to a regularly spaced grid with spatial discretisation size Δx and temporal discretisation size Δt :

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = D \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2}.$$

For ease of symbol manipulation, we have replaced the usual thermal diffusivity parameter with D . Manipulation then gives

$$u_j^{n+1} = u_j^n + \frac{Dk}{h^2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n),$$

where we have replaced Δt with k and Δx with h .

The basic idea of von Neumann analysis is to examine the growth rate when the solution is assumed to consist of sinusoidals. Let us assume that the solution at a mesh point is given by

$$u_j^n = G^n e^{i\xi x_j} = G^n e^{i\xi jh}$$

where for simplicity we have assumed that the spatial points are given by $x_j = jh$. The above is related to a sinusoidal wave with a wavenumber of ξ (assumed unspecified for now). The theory of Fourier series tells us that many functions can be considered as sums of the above sinusoidal waves, over many different values of ξ . So our task is to understand the effect of one such wave, and to examine the growth rate G .

Substitution into the finite-difference equation gives

$$G^{n+1} e^{i\xi jh} = G^n e^{i\xi jh} + \frac{Dk}{h^2}(G^n e^{i\xi (j+1)h} - 2G^n e^{i\xi jh} + G^n e^{i\xi (j-1)h}),$$

or alternatively

$$\frac{G^{n+1}}{G^n} = 1 + \frac{Dk}{h^2} (e^{i\xi h} - 2 + e^{-i\xi h}).$$

However, we can use the fact that

$$\cos(\xi h) = \frac{e^{i\xi h} + e^{-i\xi h}}{2},$$

and therefore

$$\frac{G^{n+1}}{G^n} = 1 + 2 \frac{Dk}{h^2} (\cos(\xi h) - 1).$$

Moreover, using another trigonometric identity, we can write

$$\frac{G^{n+1}}{G^n} = 1 - 4 \frac{Dk}{h^2} \sin^2(\xi h/2) \equiv g(\xi, h, k).$$

This is essentially a typical recurrence relation for G^n with a growth factor $g(\xi, h, k)$. In order for the scheme to stable over all possible wavenumbers ξ , we must then require that $|g| < 1$ (otherwise $G^n \rightarrow \infty$ as $n \rightarrow \infty$). Thus we require that

$$|1 - 4 \frac{Dk}{h^2}| < 1.$$

or

$$k < \frac{h^2}{2D} \implies \Delta t < \frac{\Delta x^2}{2D}.$$

CHAPTER 43

Problem class 5

This problem class will be devoted to getting started on problems from Problem Set 4 Chapter 28 and Problem Set 5 Chapter 29, and also to review any outstanding problems in previous sets.

CHAPTER 44

Problem class 6

This problem class will be devoted to the study of a Newton solver for the latitude-dependent EBM, which makes up Chapter 17, and then also reviewing any outstanding problem set questions.

Part VIII

Appendices

CHAPTER 45

Differential equations

45.1 First-order linear differential equations

First-order linear differential equations are of the form

$$y'(x) + q(x)y(x) = r(x).$$

These are solved by integrating factors. One trick often employed us to think of placing the left hand-side of the equation into the form:

$$\frac{d}{dx} [y(x)e^{??}],$$

where the derivative of ?? yields $q(x)$.

Officially, to do this, we multiply both sides of the original equation by the integrating factor

$$e^{\int^x q(x') dx'}$$

so that the equation can be placed in the form

$$\frac{d}{dx} [y(x)e^{\int^x q(x') dx'}] = r(x)e^{\int^x q(x') dx'}$$

Then integrate and solve for y . This gives

$$y(x) = \left(\int r(x)e^{\int^x q(x') dx'} dx + C \right) e^{-\int^x q(x') dx'}.$$

It is usually much easier to work with specific problems, as many of the integrals are simpler than the general form given above. You will need to impose any required boundary conditions in order to determine the constant of integration above.

The solution of first-order linear equations is a standard problem in most first differential equations courses. You can find many references online and here is a good one providing plenty practice: [Paul's online notes](#)

45.2 Second-order constant coefficient ODEs

Second-order linear constant-coefficient ODEs are of the form

$$y'' + ay' + by = f(x).$$

With both forced ($f \neq 0$) and unforced ($f = 0$) varieties studied. For the unforced variant (homogeneous), you will remember that the equation is solved by attempting the ansatz $y = e^{rx}$ and solving for r .

There are many references on this topic since it is the standard second-order theory learned in most initial differential equations courses.

References:

- [Paul's online notes](#)

CHAPTER 46

Dynamical systems

Appendices

These appendices may contain material that is added during the term, dependent on student enquiries and need.

We will need some of the theory from your prior ODEs and Modelling and Dynamics courses as related to the solution of 2x2 linear systems of the form

$$\mathbf{Y}'(t) = \mathbf{A}(t)\mathbf{Y},$$

where

$$\mathbf{Y}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}, \quad \text{and} \quad \mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

If you need a refresher, refer to your old notes; there is also a good review on [these notes](#) and also [these notes](#).

46.1 Stability analysis for 2x2 systems

In the case of 2x2 systems, there is a useful classification and shortcut. The eigenvalues, λ_1 and λ_2 are given by

$$\lambda_{1,2} = \frac{1}{2}(T \pm \sqrt{T^2 - 4D}),$$

where $T = \text{tr}(A) = a + d$ and $D = \det(A) = ad - bc$. Once T and D are known, then the fixed points can be classified based on the following diagram.

Above, the critical parabola is where $T^2 - 4D = 0$.

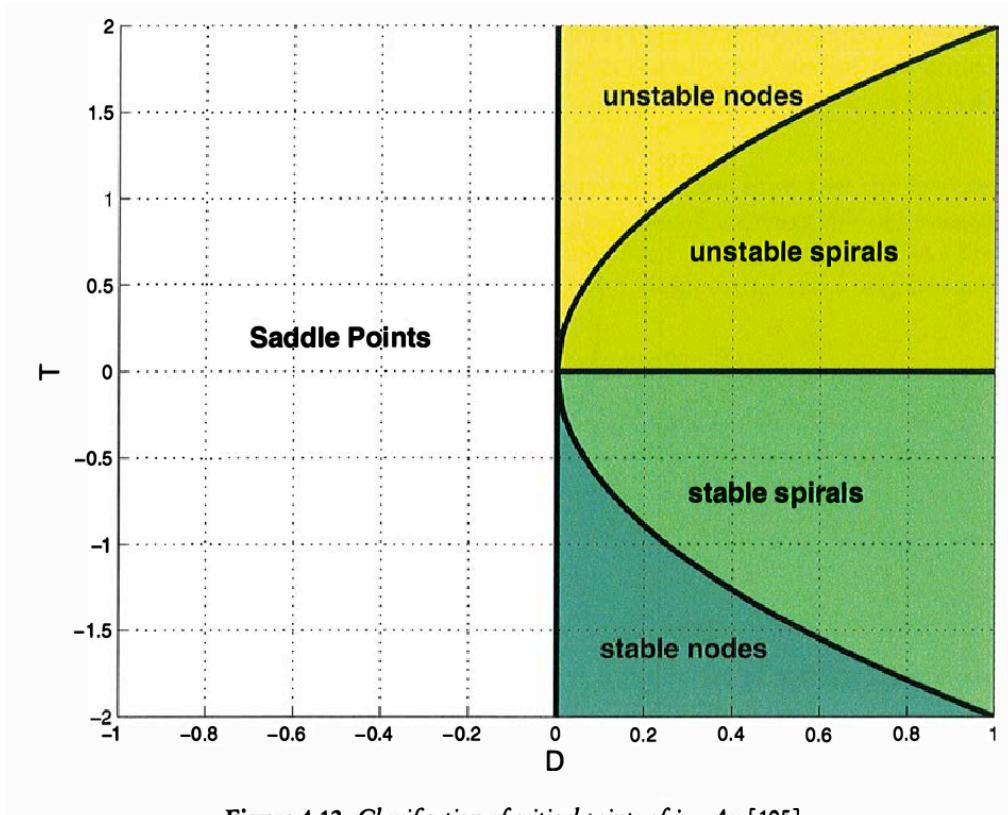


Figure 4.12. Classification of critical points of $\dot{x} = Ax$ [105].

Figure 46.1: Classification diagram from (Kaper und Engler, 2013)

CHAPTER 47

Vector calculus

Appendices

These appendices may contain material that is added during the term, dependent on student enquiries and need.

47.1 Surface integrals

Surface integrals are generalisations of the concept of a multiple integrals to integrals over surfaces. For example, consider a flat plate of area A heated to a certain temperature, and thus emits a certain amount of energy, E , per unit area (per unit time). The total energy emitted per unit time would then be EA .

If we want to consider the same problem, but now posed on a general surface, say the planet S , and where E varies along the surface, then we must consider adding together $E(\mathbf{x})dS$. We thus chop our surface into smaller pieces, each with area dS , multiply each piece with its corresponding energy, and then sum the result.

The result is the surface integral

$$\iint_S E(\mathbf{x}) dS.$$

A tutorial for computing surface integrals can be found [here](#).

You will not need to know how to calculate surface integrals in general in the course.

CHAPTER 48

Modelling

Appendices

These appendices may contain material that is added during the term, dependent on student enquiries and need.

49

CHAPTER

Finite difference approximations

Appendices

These appendices may contain material that is added during the term, dependent on student enquiries and need.

Many of the numerical methods we need will require using numerical approximations for the derivatives of functions. For instance, using Newton's method to solve for the zeros of a system of equations requires calculating the Jacobian matrix.

49.1 Forwards, backwards, and centred differences

We want to learn how to use finite differences in order to approximate derivatives numerically. We know that

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h},$$

provided the limit exists. Therefore, a simple idea to approximating the value of $f'(x)$ is to use a small numerical value for h , and calculate (the gradient of the secant line),

$$f'(x) \approx \frac{f(x + h) - f(x)}{h},$$

where h is a specified small number. The above is known as the **two-point forward-difference formula**. In fact, we can determine exactly the error of such an approximation via Taylor's theorem. If f is twice continuously differentiable, then

$$f(x + h) = f(x) + hf'(x) + \frac{h^2}{2}f''(c),$$

for some point $c \in [x, x + h]$. Therefore by re-arrangement, we see the following.

Two-point forward-difference formula

$$f'(x) = \frac{f(x + h) - f(x)}{h} - \frac{h}{2}f''(c),$$

where $c \in (x, x + h)$.

Notice that the error in using the two-point forward difference approximation is then $\mathcal{O}(h)$, and this error tends to zero as $h \rightarrow 0$ (as long as f'' is continuous). We thus call the above formula a first-order finite-difference approximation. If the error is $\mathcal{O}(h^n)$, we call the corresponding formula an n th-order approximation.

Example

Use the two-point forward difference formula with different values of h in order to approximate the derivative of $f(x) = 1/x$ at $x = 2$.

$f(x)$	$f(x + h)$	h	$f'(x)$	Error
...
...
...

Similar formulae can be developed for the backwards difference (send $h \rightarrow -h$).

A more accurate formula can be developed via subtracting the Taylor series for $f(x-h)$ from that for $f(x+h)$. This results in:

Three-point centered-difference formula

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6} f'''(c),$$

where $c \in (x-h, x+h)$.

Thus we see that the centered difference formula is accurate to $O(h^2)$.

49.2 Jacobian matrices

Reference

You will have encountered the Jacobian, firstly in your first-year Methods courses when performing change-of-coordinates in integration formulae, and secondly in your second-year Modelling and Dynamics courses when studying differential equations. It also comes up in the second-year Vectors and PDEs course.

The Jacobian matrix, \mathbf{J} , of a vector function, $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^m$,

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{pmatrix},$$

is the matrix of all its first-order partial derivatives,

$$\mathbf{J}(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \nabla f_1(\mathbf{x})^T \\ \vdots \\ \nabla f_n(\mathbf{x})^T \end{pmatrix}.$$

We can essentially think of the Jacobian as the multi-dimensional extension of the basic derivative. It encodes all of the (first-order) information about the rate-of-change of the function.

It is interesting and important to consider the numerical evaluation of the Jacobian, in the event that the function F cannot be easily differentiated exactly. The simplest algorithm is based on approximating each entry of the matrix by a finite difference.

For example, let us consider approximating

$$\frac{\partial f_i}{\partial x_j}(\mathbf{x}_0).$$

We define the j th unit vector by $\mathbf{e}_j = [0 \ 0 \ \dots \ 1 \ \dots \ 0]$, which the j th entry being one and all entries being zero. Then we can approximate the derivative by a central difference,

$$\frac{\partial f_i}{\partial x_j}(\mathbf{x}_0) \approx \frac{f_i(\mathbf{x}_0 + \mathbf{e}_j h) - f_i(\mathbf{x}_0 - \mathbf{e}_j h)}{2h},$$

where h is a small step size.

Let us test this by calculating the Jacobian for

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} x_1^2 \\ x_2^2 \\ x_1 x_2 \end{pmatrix}.$$

The pseudocode for this looks like this

Pseudocode for numerical Jacobian

Input: function F (size m), point x (size n), step size h

1. Create an $(m \times n)$ matrix, J
2. a. Loop through all the rows indexed by i
 - b. Loop through all the columns indexed by j
 - Create the \mathbf{e}_j unit vector
 - Calculate the difference of $F_i(x + h \mathbf{e}_j) - F_i(x - h \mathbf{e}_j)$
 - Divide by $2h$
 - Assign this value to the (i, j) th value of the Jacobian

Output: the $(m \times n)$ Jacobian matrix J

Here is a code that puts it into practice.

```
import numpy as np

def jacobian(func,initial,delta=1e-3):
    f = func
    nrow = len(f(initial))
    ncol = len(initial)
    output = np.zeros(nrow*ncol)
    output = output.reshape(nrow,ncol)
    for i in range(nrow):
        for j in range(ncol):
            ej = np.zeros(ncol)
            ej[j] = 1
            dij = (f(initial+ delta * ej)[i] - f(initial- delta * ej)[i])/(2*delta)
            output[i,j] = dij
```

```
    return output

def myf(x):
    x1 = x[0]
    x2 = x[1]
    output = np.zeros(3)
    output[0] = x[0]**2
    output[1] = x[1]**2
    output[2] = x[0]*x[1]
    return output

jacobian(myf,[1,2])

array([[2., 0.],
       [0., 4.],
       [2., 1.]])
```

CHAPTER 50

Coding

50.1 Variable scope

During the 27 Feb 2024 lecture, the question of variable scope came up. Firstly, here is what ChatGPT says:

The screenshot shows a dark-themed chat interface between a user (PH) and ChatGPT. The user asks why Python knows the value of 'a' within a function definition if it doesn't have local scope. ChatGPT responds by explaining lexical scoping and closures, referencing the provided code example where 'a' is defined at the top level and its value is printed from within a nested function.

You
In the following code, why does Python know the value of 'a' within the function definition? Doesn't 'a' have local scope?

```
a = 1
def myfunc():
    print(a)
```

ChatGPT
In Python, when you reference a variable within a function, if that variable is not defined locally within the function, Python will look for it in the enclosing scopes. This behavior is known as "lexical scoping" or "closure".
In your example, since the variable 'a' is not defined locally within the function `myfunc()`, Python looks for it in the enclosing scope, where it finds the variable 'a' defined with a value of 1. Therefore, when you call `myfunc()`, it prints the value of 'a' from the enclosing scope, which is 1.
So, yes, `a` has global scope in this context, and Python is able to access its value within the function `myfunc()`.

Figure 50.1: What ChatGPT says on scope

So indeed it looks like Python uses a scheme where if a variable is not locally declared, it will use values of the variable outside of the function. This can allow us to produce this kind of thing:

```
def myfunc():
    print(a)

a = 1
myfunc()
```

```
a = 2
myfunc()
a = 3
myfunc()
```

1
2
3

So I was wrong! Python variables are global when defined outside the function! [Here is a reference](#).

This is certainly easier to use! However it can lead to confusion because for very complex codes, you may not be sure what value of a variable is being used within a function. Because most of the programming you will do in this course is very simple, this is unlikely to lead to trouble.

It's generally good to avoid this whenever it is possible, though. For instance in this particular case, it makes much more sense to pass in the variable.

```
def myfunc(a):
    print(a)

a = 99
myfunc(1)
myfunc(2)
myfunc(3)
```

1
2
3

In the above case, the `a` that appears in the function is different from the `a` that is declared outside the function. This makes it very easy to understand what your function is doing without knowing what is happening outside of it.

For example, there is no danger of values being reset:

```
def myfunc(a):
    a = a + 1000
    print(a)

a = 99
myfunc(1)
myfunc(2)
myfunc(3)
print(a)
```

1001
1002
1003
99

50.2 Conversations with ChatGPT

We should have an informal discussion of how the use of GenAI in education. Does it help with coding? When should it shoudn't it be used?

Asking something simple.

You
Write me a python code to solve an advection diffusion equation

ChatGPT
Sure, here's a simple Python code using finite difference method to solve a 1D advection-diffusion equation:

```
python
import numpy as np
import matplotlib.pyplot as plt

# Parameters
L = 1.0 # Length of the domain
nx = 100 # Number of grid points
dx = L / (nx - 1) # Grid spacing
nt = 100 # Number of time steps
dt = 0.01 # Time step size
u = 1.0 # Advection velocity
D = 0.1 # Diffusion coefficient

# Initial condition
x = np.linspace(0, L, nx)
u_initial = np.sin(np.pi * x)

# Explicit finite difference method
u_new = np.copy(u_initial)
for n in range(1, nt):
    u_old = np.copy(u_new)
    for i in range(1, nx - 1):
        u_new[i] = u_old[i] + dt * (D * (u_old[i+1] - 2*u_old[i] + u_old[i-1])) / dx**2

# Plotting
plt.plot(x, u_initial, label='Initial Condition')
plt.plot(x, u_new, label='Final Solution')
plt.xlabel('x')
plt.ylabel('u')
plt.title('Advection-Diffusion Equation Solution')
plt.legend()
plt.grid(True)
plt.show()
```

This code sets up a 1D domain, discretizes it using finite differences, and evolves the solution in time using an explicit finite difference scheme. Finally, it plots the initial and final solutions. You can adjust the parameters ('L', 'nx', 'nt', 'dt', 'u', 'D') to suit your specific problem.

Figure 50.2: A nice chat with ChatGPT

 You

On the previous code, how does it treat the boundaries?

 ChatGPT

In the previous code, the boundaries are treated implicitly. The finite difference method is applied to the interior points of the domain, and the boundary points are not updated explicitly within the time-stepping loop.

For example, in the line:

 python Copy code

```
for i in range(1, nx - 1):
```

The loop iterates from the second grid point (``i=1``) to the second-to-last grid point (``i=nx-2``). This means the first and last grid points, which correspond to the boundaries of the domain, are not updated within this loop.

However, it's important to note that this code does not include boundary conditions explicitly. Depending on the specific problem being solved, appropriate boundary conditions would need to be applied separately. These boundary conditions could be Dirichlet, Neumann, or periodic boundary conditions, depending on the nature of the problem.

Wait a minute... |

 **You**
Modify the previous code to use dirichlet boundary conditions.

 **ChatGPT**
To modify the code to use Dirichlet boundary conditions, we'll need to specify the values of the function at the boundary points and ensure that these values are not updated during the time-stepping loop. Here's the modified code:

```
python Copy code  
  
import numpy as np  
import matplotlib.pyplot as plt  
  
# Parameters  
L = 1.0 # Length of the domain  
nx = 100 # Number of grid points  
dx = L / (nx - 1) # Grid spacing  
nt = 100 # Number of time steps  
dt = 0.01 # Time step size  
u = 1.0 # Advection velocity  
D = 0.1 # Diffusion coefficient  
  
# Initial condition  
x = np.linspace(0, L, nx)  
u_initial = np.sin(np.pi * x)  
  
# Dirichlet boundary conditions  
u_left = 0.0  
u_right = 0.0  
  
# Explicit finite difference method with Dirichlet boundary conditions  
u_new = np.copy(u_initial)  
for n in range(1, nt):  
    u_old = np.copy(u_new)  
    for i in range(1, nx - 1):  
        u_new[i] = u_old[i] + dt * (D * (u_old[i+1] - 2*u_old[i] + u_old[i-1]) / dx**2)  
    # Applying Dirichlet boundary conditions  
    u_new[0] = u_left  
    u_new[-1] = u_right  
  
# Plotting  
plt.plot(x, u_initial, label='Initial Condition')  
plt.plot(x, u_new, label='Final Solution')  
plt.xlabel('x')  
plt.ylabel('u')  
plt.title('Advection-Diffusion Equation Solution with Dirichlet Boundary Conditions')  
plt.legend()
```

 You

Write a python code to solve the 1D heat equation from $x = 0$ to $x = 10$. The boundary condition at $x = 0$ is oscillating in time with $u(0, t) = \cos(t)$. Plot the solution every k timesteps

 ChatGPT

To solve the 1D heat equation with an oscillating boundary condition at $x = 0$, we can use the finite difference method. Here's the Python code to solve the equation and plot the solution every k timesteps:

python

 Copy code

```

import numpy as np
import matplotlib.pyplot as plt

# Parameters
L = 10.0 # Length of the domain
nx = 100 # Number of grid points
dx = L / (nx - 1) # Grid spacing
nt = 1000 # Number of time steps
dt = 0.01 # Time step size
k = 10 # Plotting frequency
D = 0.1 # Diffusion coefficient

# Initialize arrays
x = np.linspace(0, L, nx)
u_initial = np.zeros(nx)
u_solution = np.zeros((nt, nx))

# Function to solve heat equation using finite differences
def solve_heat_equation(u_initial):
    u_new = np.copy(u_initial)
    for i in range(1, nx - 1):
        u_new[i] = u_initial[i] + D * dt * (u_initial[i+1] - 2*u_initial[i] + u_initial[i-1])
    return u_new

# Set initial condition
u_initial[0] = 1.0 # Boundary condition at x=0

# Solve heat equation and plot solution every k timesteps
for t in range(nt):
    if t % k == 0:
        plt.plot(x, u_initial, label=f'Timestep {t}')
        u_initial = solve_heat_equation(u_initial)

# Update the boundary condition at x=0 with cos(t)
u_initial[0] = np.cos(0)

# Plot final solution
plt.plot(x, u_initial, label=f'Timestep {nt}')
plt.xlabel('x')
plt.ylabel('u')
plt.title('1D Heat Equation Solution with Oscillating Boundary Condition')
plt.legend()
plt.grid(True)
plt.show()

```

This code uses a finite difference method to solve the 1D heat equation and plots the solution every k timesteps. The boundary condition at $x = 0$ is set to oscillate as $\cos(t)$. Adjust the parameters `nt`, `dt`, and `k` as needed for your specific problem.

Modifying again to oscillate in time...

 You
Your previous solution is not updating the boundary condition at every iteration

 ChatGPT
You're correct. I apologize for the oversight. To ensure that the boundary condition at $x = 0$ is updated at every iteration, we need to update it inside the loop. Here's the corrected code:

```
python 
```

```

import numpy as np
import matplotlib.pyplot as plt

# Parameters
L = 10.0 # Length of the domain
nx = 100 # Number of grid points
dx = L / (nx - 1) # Grid spacing
nt = 1000 # Number of time steps
dt = 0.01 # Time step size
k = 10 # Plotting frequency
D = 0.1 # Diffusion coefficient

# Initialize arrays
x = np.linspace(0, L, nx)
u_initial = np.zeros(nx)
u_solution = np.zeros((nt, nx))

# Function to solve heat equation using finite differences
def solve_heat_equation(u_initial):
    u_new = np.copy(u_initial)
    for i in range(1, nx - 1):
        u_new[i] = u_initial[i] + D * dt * (u_initial[i+1] - 2*u_initial[i] + u_initial[i-1])
    return u_new

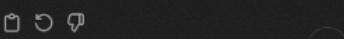
# Set initial condition
u_initial[0] = 1.0 # Boundary condition at x=0

# Solve heat equation and plot solution every k timesteps
for t in range(nt):
    if t % k == 0:
        plt.plot(x, u_initial, label=f'Timestep {t}')
    u_initial = solve_heat_equation(u_initial)
    # Update boundary condition at x=0 with cos(t)
    u_initial[0] = np.cos(t * dt)

# Plot final solution
plt.plot(x, u_initial, label=f'Timestep {nt}')
plt.xlabel('x')
plt.ylabel('u')
plt.title('1D Heat Equation Solution with Oscillating Boundary Condition')
plt.legend()
plt.grid(True)
plt.show()

```

In this corrected code, the boundary condition at $x = 0$ is updated within the time-stepping loop at every iteration using $\cos(t \cdot dt)$, where t is the current timestep. This ensures that the boundary condition oscillates with time as expected.

Oops! 

CHAPTER 51

An example of ChatGPT going wrong

Please refer to [the Courseworks and genAI document for guidance](#).

Remember that standard generative AI tools do not have mathematical engines. They have mathematical information, but no knowledge of mathematics. They can produce content that looks convincing but is incorrect. Worse still, in order to diagnose the issues, you may need to already be an expert!

Here is an example based on material from this course.

 You

Solve the differential equation $e^p y'' + y^2 = \operatorname{sech}(x)$ using matched asymptotics with $y(0) = 0$ and $y(1) = 1$

 ChatGPT

To solve the differential equation

$$e^p y'' + y^2 = \operatorname{sech}(x)$$

using matched asymptotics, we'll first identify the scales of y and x . Since we're looking for solutions near $x = 0$, we'll assume a boundary layer solution near $x = 0$ where $\operatorname{sech}(x)$ is significant. In the outer region, we'll look for solutions where y is significant.

For small x , the scale of $\operatorname{sech}(x)$ is $O(1)$, while for y , the scale is $O(\sqrt{\epsilon})$ where ϵ is a small parameter.

Let $x = \epsilon^\alpha X$, where α is to be determined later. For the inner region, we take $\epsilon \ll X \ll 1$, and for the outer region, we take $X = O(1)$. We will match the solutions in the overlap region $0 \leq X \leq O(\epsilon^\beta)$ where $\beta > \alpha$.

Substituting $x = \epsilon^\alpha X$ and $y(x) = \epsilon^\gamma Y(X)$ into the differential equation and neglecting terms of higher order in ϵ yields:

$$e^p \epsilon^{-2\alpha} Y'' + \epsilon^{2\gamma} Y^2 = \operatorname{sech}(\epsilon^\alpha X)$$

For balance, we choose:

$$2\gamma - 2\alpha = 0 \implies \gamma = \alpha$$

$$2\alpha - 2\alpha p = 0 \implies p = 1$$

Thus, we need to solve:

$$Y'' + Y^2 = e^X$$

Figure 51.1: Example ChatGPT output

Now we look for the outer solution. We seek a solution of the form $y(x) = Y_0(x) + \epsilon Y_1(x) + \dots$. Substituting into the original equation and equating powers of ϵ gives:

$$O(1) : Y_0'' + Y_0^2 = 0$$

$$O(\epsilon) : Y_1'' + Y_1^2 = 1$$

The solution for Y_0 is the outer solution, which satisfies $Y_0(0) = 0$ and $Y_0(1) = 1$.

For the inner solution, we seek a solution of the form $y(x) = y_0(X) + \epsilon y_1(X) + \dots$. Substituting into the original equation and equating powers of ϵ gives:

$$O(1) : y_0'' + y_0^2 = 0$$

$$O(\epsilon) : y_1'' + y_1^2 = e^X$$

Now we can solve these equations for Y_0 , Y_1 , y_0 , and y_1 and match them in the overlap region. Matching will require solving for α and β and ensuring continuity and smoothness of the solutions in the overlap region.

This process involves some intricate algebraic and calculus steps which I can guide you through if needed, but it's lengthy to write out in detail here. Let me know if you'd like to proceed with the actual calculations or if you need further clarification on any part.



Figure 51.2: Example ChatGPT output

It is important to remember that “correcting” the output of ChatGPT is sort of a misnomer. ChatGPT or similar genAI or Large Language Models and algorithms do not include common sense, or any algorithmic understanding of syntactic rules. With “correcting” misunderstandings, you generally need to understand why the error was made in the first place. With LLMs, the “why” is replaced by the fact that, within its training set, words are simply close together in the high-dimensional vector space.

This brings to mind the well-known phrase “*Not even wrong*”

Not even wrong

14 languages ▾

Article Talk

Read Edit View history Tools ▾

From Wikipedia, the free encyclopedia

For other uses, see [Not even wrong \(disambiguation\)](#).

"**Not even wrong**" is a phrase often used to describe [pseudoscience](#) or bad science. It describes an argument or explanation that purports to be [scientific](#) but uses [faulty reasoning](#) or speculative premises, which can be neither affirmed nor denied and thus cannot be discussed rigorously and [scientifically](#). The phrase "not even wrong" is synonymous with "unfalsifiable".^[1]

History of the expression [edit]

The phrase is generally attributed to the [theoretical physicist Wolfgang Pauli](#), who was known for his colorful objections to incorrect or careless thinking.^{[2][3]} Rudolf Peierls documents an instance in which "a friend showed Pauli the paper of a young physicist which he suspected was not of great value but on which he wanted Pauli's views. Pauli remarked sadly, 'It is not even wrong'."^{[4][5]} This is also often quoted as "That is not only not right; it is not even wrong", or in Pauli's native German, "*Das ist nicht nur nicht richtig; es ist nicht einmal falsch!*" Peierls remarks that quite a few [apocryphal stories](#) of this kind have been circulated and mentions that he listed only the ones personally vouched for by him. He also quotes another example when Pauli replied to [Lev Landau](#), "What you said was so confused that one could not tell whether it was nonsense or not."^[4]

Figure 51.3: Not even wrong

51.1 Breakdown of issues

To solve the differential equation

$$\epsilon^p y'' + y^2 = \operatorname{sech}(x)$$

Solution

Why did it change “ep” to ϵ^p ?

using matched asymptotics, we'll first identify the scales of y and x . Since we're looking for solutions near $x = 0$, we'll assume a boundary layer solution near $x = 0$ where $\operatorname{sech}(x)$ is significant.

Solution

Why did it assume the boundary layer is near $x = 0$? Also, x is not necessarily “significant” near $x = 0$.

In the outer region, we'll look for solutions where y is significant.

Solution

Doesn't really make sense.

For small x , the scale of $\operatorname{sech}(x)$ is $O(1)$, while for y , the scale is $O(\sqrt{\epsilon})$ where ϵ is a small parameter.

Solution

That's not right... x is not $O(1)$ near $y = 0$. It's not obvious at this point what the scaling for y is.

Let $x = \epsilon^\alpha X$, where α is to be determined later. For the inner region, we take $\epsilon \ll X \ll 1$, and for the outer region, we take $X = O(1)$. We will match the solutions in the overlap region $0 \leq X \leq O(\epsilon^\beta)$ where $\beta > \alpha$.

Solution

This isn't right. In the inner region $X = O(1)$ and the outer region $X \rightarrow \infty$. Almost all the statements are wrong.

Substituting $x = \epsilon^\alpha X$ and $y(x) = \epsilon^\gamma Y(X)$ into the differential equation and neglecting terms of higher order in ϵ yields:

$$e^p \epsilon^{-2\alpha} Y'' + \epsilon^{2\gamma} Y^2 = \operatorname{sech}(\epsilon^\alpha X)$$

For balance, we choose:

$$\begin{aligned} 2\gamma - 2\alpha &= 0 \implies \gamma = \alpha \\ 2\alpha - 2\alpha p &= 0 \implies p = 1 \end{aligned}$$

Thus, we need to solve:

$$Y'' + Y^2 = e^X$$

Solution

The above lines don't make any sense. For one thing, near $x = 0$, you should have $\operatorname{sech}x \sim 1$. Secondly if you examine what its working gives, no value of α is obtained. The correct inner equation is found by balancing the first and last terms, giving $x = \sqrt{\epsilon}X$ and

$$Y'' + \epsilon Y^2 = \operatorname{sech}(\epsilon x) \sim 1.$$

Everything else is wrong.

References

- [Budyko 1969] BUDYKO, Mikhail I.: The effect of solar radiation variations on the climate of the Earth. In: *tellus* 21 (1969), Nr. 5, S. 611–619
- [Fourier 1827] FOURIER, Joseph: Mémoire sur les températures du globe terrestre et des espaces planétaires. In: *Mémoires de l'Académie Royale des Sciences de l'Institut de France* 7 (1827), S. 570–604
- [Fowler 2011] FOWLER, Andrew: *Mathematical geoscience*. Bd. 36. Springer, 2011
- [Kaper und Engler 2013] KAPER, H. ; ENGLER, H.: *Mathematics and climate*. SIAM, 2013
- [MacKay 2009] MACKAY, D.J.C.: *Sustainable energy – without the hot air*. UIT Cambridge Ltd., 2009
- [Pouillet 1838] POUILLET, Claude Servais M.: Memoire sur le chaleur solaire. In: *Paris* (1838)
- [Stommel 1961] STOMMEL, Henry: Thermohaline convection with two stable regimes of flow. In: *Tellus* 13 (1961), Nr. 2, S. 224–230
- [Van der Veen 2000] VEEN, CJ Van der: Fourier and the “greenhouse effect”. In: *Polar Geography* 24 (2000), Nr. 2, S. 132–152