Numerical methods for complex systems

SoSe 09

Institute for Theoretical Physics, WWU Münster

July 22, 2009

Contents

Chapter 1

Introduction to Spectral Methods

1.1 Basic Concepts

Our starting point is a basic question. Suppose we have an equation for some vector function $u(x)$, $x \in \Omega \subseteq \mathbb{R}^n$

$$\mathcal{L}u = f, \tag{1.1}$$

with boundary conditions

$$\mathcal{B}u = 0, \qquad x \in \partial\Omega, \tag{1.2}$$

where $\mathcal{L}$ and $\mathcal{B}$ are some linear operator. How can we find the best approximation of the unknown function u? One of possible methods is based on the wide class of discretization schemes known as *method of weighted residuals (MWR)*. The idea of the method is to approximate the unknown function $u(x)$ by a sum of so-called *trial* or *basis* functions $\phi_n(x)$

$$\widetilde{u}(x) = \sum_{n=0}^{N} a_n \phi_n(x), \tag{1.3}$$

where a_n are unknown coefficients to be determined and the tilde denotes an approximate solution. If one substitute the approximation (1.3) into Eq. (1.1), *the residual* R can be calculated as

$$R = \mathcal{L}\widetilde{u} - f \tag{1.4}$$

Because $\widetilde{u}$ is different from the exact solution, the residual does not vanish for all $x \in \Omega$. The next step is to determine unknown a_n such that the

chosen function best approximates the exact solution. To this end, a *test* or *weighting* functions $\chi_n(x)$, $n = 0, \ldots, N$ are selected so that the residual function is minimized, e.g., the weighted average of the residual over the domain of interest is set to zero,

$$\int_\Omega \chi_n(x) R \, dx = 0 \qquad n = 0, \ldots, N. \tag{1.5}$$

The various methods differ mainly in the choise of trial and test functions and their minimization strategies.

Various numerical methods

The choice of trial functions ϕ_n is one of the key difference between finite-element and finite-difference methods respectively, and spectral methods. In the case of finite-element methods the domain Ω is divided into small finite intervals and ϕ_n are typically choosen to be a *local* polynomial of fixed degree, defined on these sub-intervals only. The finite-difference methods have a local charachter as well. Generally, the unknown function $u(x)$ is approximated by a sequence of overlapping polynomials of low order, interpolating the solution at a given set of discretization points and the rersult is represented in the form of weighted sum of values of $u(x)$ at the interpolation points. In contrast, the trial functions for spectral methods are infinitely differentiable *global* functions, e.g., Fourier or Chebyshev series. The particular choice of the trial functions is usually connected with the geometry of the problem. For instance, on periodic intervals, the sines and cosines of a Fourier series, which automatically satisfy boundary conditions are used. For non-periodic problem, Chebyshev or Legendere polynomials are more natural choise.

Various minimization strategies

The choice of the test functions χ_n distinguishes between the three most commonly used spectral shemes, namely

1. *Galerkin method.*

   The test functions χ_n are the same as the trial functions and each ϕ_n satisfy the boundary condition $\mathcal{B}\phi_n = 0$. Since $\phi_n = \chi_n$ for $n =$

$0, \ldots, N$, Eq. (1.5) is equivalent to

$$\int_\Omega \phi_n R = 0 \Leftrightarrow \int_\Omega \phi_n \big(\mathcal{L}\widetilde{u} - f\big) = 0 \Leftrightarrow \int_\Omega \phi_n \sum_{k=0}^{N} a_k \phi_k = \int_\Omega \phi_n f \Leftrightarrow$$

$$\sum_{k=0}^{N} L_{nk} a_k = \int_\Omega \phi_n f,$$

where $L_{nk} = \int_\Omega \phi_n \mathcal{L}\phi_k$. Solving the obtained linear system one can get all $N+1$ unknown coefficients a_k.

2. *Tau method.* The test functions χ_n are the same as the trial functions, but ϕ_n do not need to satisfy (1.2). The boundary conditions are enforced by an additional set of equations. In order to find this set, let us consider an orthonormal basis $\{\psi_l\}$, $l = 0, \ldots, M$, where $M < N$ on the $\partial\Omega$ and expand $\mathcal{B}\phi_n$ upon it:

$$\mathcal{B}\phi_n = \sum_{l=0}^{M} B_{ln} \psi_l.$$

Equation (1.2) then becomes

$$\mathcal{B}u = 0 \Leftrightarrow \sum_{k=0}^{N} \sum_{l=0}^{M} a_k B_{lk} \psi_l = 0 \Leftrightarrow \sum_{k=0}^{N} B_{lk} a_k = 0, \quad l = 0, \ldots, M.$$

The resulting linear system of $N+1$ equations consists of $N-M$ first rows of the Galerkin system, presented above, and $M+1$ additional equations for boundary conditions:

$$\sum_{k=0}^{N} L_{nk} a_k = \int_\Omega \phi_n f, \qquad n = 0, \ldots, N-M-1,$$

$$\sum_{k=0}^{N} B_{lk} a_k = 0, \qquad l = 0, \ldots, M.$$

3. *Collocation (pseudospectral) method.* The test functions are represented by a delta functions at special points x_n, called *collocation points*, i.e., $\chi_n = \delta(x - x_n)$. In other words this approach requires Eq. (1.1) to be satisfied exactly at the collacation points x_n. Then the condition (1.5)

reads:

$$\int_\Omega \phi_n R = 0 \Leftrightarrow \int_\Omega \delta(x-x_n)R = 0 \Leftrightarrow \mathcal{L}\widetilde{u}(x_n) = f(x_n) \Leftrightarrow$$
$$\sum_{k=0}^{N} a_k \mathcal{L}\phi_k(x_n) = f(x_n), \qquad n = 0, \ldots, N.$$

The boundary conditions can be imposed as in the tau method.

To sum up: Galerkin and tau methods are implemented in terms of the expansion coefficients, whereas the collocation method is implemented in terms of the physical space values of the unknown function.

1.2 Fourier and Chebyshev series

1.2.1 The Fourier System

A Fourier series is an expansion of a periodic function in terms of an infinite sum of sines and cosines. Consider a periodic integrable function $f(x)$, $x \in [-\pi, \pi]$. The numbers

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\cos(nt)\,dt, \qquad n \geq 0,$$
$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\sin(nt)\,dt, \qquad n \geq 1$$

are called *the Fourier coefficients* of f. The *Fourier series* of a function $f(x)$ is given by

$$\boxed{f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\Big(a_n\cos(nx) + b_n\sin(nx)\Big).}$$

If the function $f(x)$ is periodic on some interval $[-L, L]$, a simple change of variables

$$x' = \frac{xL}{\pi}$$

can be used to transform the interval of integration. In this case the Fourier series read

$$f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\left(a_n\cos(\frac{n\pi x'}{L}) + b_n\sin(\frac{n\pi x'}{L})\right)$$

with

$$a_n = \frac{1}{L}\int_{-L}^{L} f(x')\cos(\frac{n\pi x'}{L})\,dx', \qquad n \geq 0,$$
$$b_n = \frac{1}{L}\int_{-L}^{L} f(x')\sin(\frac{n\pi x'}{L})\,dx', \qquad n \geq 1$$

One of the main questions is to decide when Fourier series converge, and when the sum is equal to the original function. If a function is *square-integrable* on the interval $[-\pi, \pi]$, then the Fourier series converges to the function *at almost every point*. In particular, the Fourier series converges *absolutely* and *uniformly* to $f(x)$ whenever its derivative is square integrable.

A piecewise regular function that has a finite number of finite discontinuities and a finite number of extrema can be expanded in a Fourier series which converges to the function at continuous points and the mean of the positive and negative limits at points of discontinuity (Dirichlet Conditions). As a result, near points of discontinuity, the so-called *Gibbs phenomenon*, discussed below, occurs.

Gibbs phenomenon

The Gibbs phenomenon is an overshoot (or "ringing") of Fourier series and other eigenfunction series occurring at simple discontinuities: the n'th partial sum of the Fourier series has large oscillations near the jump, which might increase the maximum of the partial sum above that of the function itself (see Fig.).

Exponential Fourier series

The notion of a Fourier series can also be extended to complex coefficients. Consider a real-valued function $f(x)$. Then using Euler's formula we can write

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx},$$

where Fourier coefficients are given by

$$c_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)e^{-inx}\,dx.$$

The Fourier coefficients a_n and b_n can be found as

$$a_n = c_n + c_{-n}, \qquad n = 0,\, 1,\, 2, \ldots,$$
$$b_n = i(c_n - c_{-n}), \qquad n = 1,\, 2, \ldots.$$

For a function periodic in $[-L,\, L]$, these become

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{\frac{in\pi x}{L}}$$

with

$$c_n = \frac{1}{2L}\int_{-L}^{L} f(x) e^{-\frac{in\pi x}{L}} dx.$$

These equations are the basis for *the Fourier transform*, which is obtained by transforming c_n from a discrete variable to a continuous one as the length $L \to \infty$.

Fourier Transformation

The Fourier transform can be considered as a generalization of the complex Fourier series in the limit $L \to \infty$. Replacing the discrete coefficient c_n with the continuous $F(k)dk$, $n/L \to k$ and changing the sum to an integral one obtains for an integrable function $f(x)$

$$f(x) = \int_{-\infty}^{\infty} F(k) e^{2\pi i k x} dk, \tag{1.6}$$

$$F(k) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} dx. \tag{1.7}$$

Here,

$$\boxed{F(k) = \mathcal{F}[f(x)](k) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} dx} \tag{1.8}$$

is called *the forward Fourier transform*, and

$$\boxed{f(x) = \mathcal{F}^{-1}[F(k)](x) = \int_{-\infty}^{\infty} F(k) e^{2\pi i k x} dk} \tag{1.9}$$

is called *the inverse Fourier transform*. However, other notation can also be found in the literature. Especially physicists prefer to write the transform, presented above in terms of the angular frequency ω:

$$F(\omega) = \mathcal{F}[f(t)](\omega) \;:=\; \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt, \tag{1.10}$$

$$f(t) = \mathcal{F}^{-1}[F(\omega)](t) \;:=\; \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega \tag{1.11}$$

Basic properties of the Fourier transform: Let us consider two integrable functions $f(x)$ and $g(x)$. Then

- *Linearity:* For any complex numbers α and β

$$\mathcal{F}[\alpha f(t) + \beta g(t)] = \alpha \mathcal{F}[f(t)] + \beta \mathcal{F}[g(t)];$$

- *Convolution:*

$$\begin{aligned} \mathcal{F}[f(t) \cdot g(t)] &= \mathcal{F}[f(t)] * \mathcal{F}[g(t)], \\ \mathcal{F}[f(t) * g(t)] &= \mathcal{F}[f(t)] \cdot \mathcal{F}[g(t)]; \end{aligned}$$

- *Translation:* For any real t_0

$$\mathcal{F}[f(t - t_0)] = e^{-i\omega t_0} \mathcal{F}[f(t)];$$

- *Scaling:*For all non-zero real numbers a

$$\mathcal{F}[f(at)] = \frac{1}{|a|} F\left(\frac{\omega}{a}\right);$$

- *Derivative:*

$$\mathcal{F}\left[\frac{d^n}{dt^n} f(t)\right] = (i\omega)^n \mathcal{F}[f(t)].$$

Examples:

1. Fourier Transform–Gaussian. Consider $f(t) = e^{-at^2}$, $\mathrm{Re}(a) > 0$. Then

$$\mathcal{F}[f(t)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-at^2} (\cos(\omega t) - i \sin(\omega t))\, dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-at^2} \cos(\omega t)\, dt = \frac{1}{2\sqrt{a}} e^{-\omega^2/4a}$$

. This shows that the Gaussian function is its own Fourier transform for some choice of a.

2. Fourier Transform–Cosine. Consider $f(t) = \cos(at)$. Then one gets:

$$\mathcal{F}[f(t)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega t} \left(\frac{e^{iat} + e^{-iat}}{2}\right) dt = \sqrt{2\pi} \left(\frac{\delta(\omega - a) + \delta(\omega + a)}{2}\right).$$

Discrete Fourier Transformation

Now consider generalization to the case of a discrete function. The sequence of N complex numbers $x_0, ..., x_{N-1}$ is transformed into the sequence of N complex numbers $X_0, ..., X_{N-1}$ by *the discrete Fourier transformation (DFT)* according to the formula:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn}, \qquad k = 0, \ldots, N-1. \tag{1.12}$$

The inverse discrete Fourier transform (IDFT) is defined as

$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N}kn}, \qquad n = 0, \ldots, N-1. \tag{1.13}$$

The DTF (1.12) is a linear transformation, so one can condiser it as a transformation of the vector $x = (x_0, x_1, \ldots, x_{N-1})^T$ to the vector $X = (X_0, X_1, \ldots, X_{N-1})^T$ of the same lenght via the relation

$$X = \hat{A}x,$$

where

$$\hat{A} = \begin{pmatrix} 1 & 1 & 1 & 1 & \ldots & 1 \\ 1 & e^{-\frac{2\pi i}{N}} & e^{-\frac{4\pi i}{N}} & e^{-\frac{6\pi i}{N}} & \ldots & e^{-\frac{2\pi i}{N}(N-1)} \\ 1 & e^{-\frac{4\pi i}{N}} & e^{-\frac{8\pi i}{N}} & e^{-\frac{12\pi i}{N}} & \ldots & e^{-\frac{2\pi i}{N}2(N-1)} \\ 1 & e^{-\frac{6\pi i}{N}} & e^{-\frac{12\pi i}{N}} & e^{-\frac{18\pi i}{N}} & \ldots & e^{-\frac{2\pi i}{N}3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-\frac{2\pi i}{N}(N-1)} & e^{-\frac{2\pi i}{N}2(N-1)} & e^{-\frac{2\pi i}{N}3(N-1)} & \ldots & e^{-\frac{2\pi i}{N}(N-1)^2} \end{pmatrix} \tag{1.14}$$

If $x_0, \ldots, x_{N-1}$ are real numbers then the DFT (1.12) obeys the symmetry:

$$X_k = \overline{X}_{N-k},$$

where the overline denotes complex conjugation. The subscripts are interpreted modulo N. As a result of the above relation, a periodic function will contain transformed peaks in not one, but two places. This happens because the periods of the input data become split into "positive" and "negative" frequency complex components. Therefore, the DFT output for real inputs is half redundant, and one obtains the complete information by only looking at roughly half of the outputs $X_0, \ldots, X_{N-1}$. The next point is that the component X_0 is always real for real data. The DFT can be computed efficiently in practice using *a fast Fourier transform (FFT)* algorithm.With a FFT, the resulting algorithm takes $\mathcal{O}(NlogN)$ arithmetic operations instead of $\mathcal{O}(N^2)$. The most common FFT are based on the co-called *Cooley-Tukey algorithm.* The most well-known use of the Cooley-Tukey algorithm is to divide the transform into two pieces of size $N/2$ at each step, and is therefore limited to power-of-two sizes , but any factorization can also be used.

Aliasing

Aliasing refers to an effect that causes different continuous signals to become indistinguishable (or aliases of one another) when sampled. Two complex exponentials $f_1 = \exp(ik_1x)$ and $f_2 = \exp(ik_2x)$ are different over $\mathbb{R}$ for different k_1 and k_2. If we restrict both function to be discrete on some finite interval, they take values $f_{1j} = \exp(ik_1x_j)$ and $f_{2j} = \exp(ik_2x_j)$, and $f_{1j} = f_{2j}$ for each j, if $k_1 - k_2 = a\frac{2\pi}{N}$, $a \in \mathbb{Z}$.

Aliasing can be caused by discrete sampling below *the Nyquist frequency.* The Nyquist frequency (or the cut-off frequency) is half the sampling frequency of a discrete signal processing system,

$$f_{\text{nyquist}} = \frac{f_{\text{sample}}}{2}$$

The sampling theorem shows that aliasing can be avoided if the Nyquist frequency is greater than the bandwidth, or maximum component frequency, of the signal being sampled.

1.2.2 An example: The Advection Equation

Let us consider a one-dimensional advection equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \tag{1.15}$$

Here $u = u(x,t)$, $x \in \mathbb{R}$, c is a nonzero constant velocity. Equation (1.15) describes the motion of a scalar u as it is advected by a known velocity field. The unique solution of (1.15) is determined by initial condition $u(x,0) = u_0$, where $u_0 = u_0(x)$ is a given fuction defined onto $\mathbb{R}$

$$u(x,t) = u_0(x - ct). \tag{1.16}$$

Solution (1.16) is just an initial function u_0 shifted by ct to the right (for $c > 0$) or to the left ($c < 0$).

Our goal is to solve Eq. (1.15) on the periodic domain $x \in [0,\, 2\pi]$ with periodic boundary conditions, i.e., $u(0,\, t) = u(2\pi,\, t)$ by means of the Galerkin method. First of all we rewrite Eq. (1.15) in the weak form, i.e., for any test function $\chi(x,\, t)$

$$\langle \partial_t u | \chi \rangle + c\langle \partial_x u | \chi \rangle = 0,$$

where $\langle f, g \rangle \equiv \int_0^{2\pi} f(x)\overline{g(x)}\, dx$ following inner product notation. Choosing the trigonometric polynomial as the trial functions, $\phi_k(x) = \exp(ikx)$, the

approximated solution $\widetilde{u}$ of 1.15 is represented as

$$\widetilde{u}(x,t) = \sum_{k=-N/2}^{N/2} \widehat{u}_k(t) e^{ikx},$$

where $\widehat{u}_k(t) = \frac{1}{2\pi}\langle \widetilde{u} | e^{ikx} \rangle$.
According to the Galerkn method (see section (1.1)) the trial functions ϕ_k and the test functions χ are essentialy the same. This reduces the problem to

$$\langle \partial_t \widetilde{u} | e^{ikx} \rangle + c \langle \partial_x \widetilde{u} | e^{ikx} \rangle = 0, \qquad \forall t > 0, \forall k = -N/2, \ldots, N/2.$$

Using the orthogonality relation $\langle e^{ilx} | e^{ikx} \rangle = 2\pi\delta_{lk}$, where δ_{lk} is the Kronecker delta, we simplify the relation above for each k to

$$\langle \partial_t \sum_{l=-N/2}^{N/2} \widehat{u}_l(t) e^{ilx} | e^{ikx} \rangle + c \langle \partial_x \sum_{l=-N/2}^{N/2} \widehat{u}_l(t) e^{ilx} | e^{ikx} \rangle = 0 \Leftrightarrow$$

$$\boxed{\frac{d\widehat{u}_k(t)}{dt} + ikc\,\widehat{u}_k(t) = 0, \qquad \forall t > 0, \forall k = -N/2, \ldots, N/2.}$$

With Fourier transformed initial conditions $\widehat{u}_k(0) = \frac{1}{2\pi}\langle u0(x) | e^{ikx} \rangle$ this coupled system of ordinary differential equations involves a standart initial value problem and can be integrated in time (using, e.g., a Runge–Kutta technique (see Appendix (.1))) to find a solution.

1.2.3 The Chebyshev system

Chebyshev polynomials

The Chebyshev (Tschebyschow, Tschebyschew) polynomials are a sequence of orthogonal polynomials which are easily to define recursively. One usually distinguishes between Chebyshev polynomials of the first kind $\{T_n\}_{n=0}^{\infty}$ and Chebyshev polynomials of the second kind $\{U_n\}_{n=0}^{\infty}$. In the study of differential equations they arise as the solution of the so-called *Chebyshev differential equations*

$$(1 - x^2)\, y'' - x\, y' + n^2\, y = 0,$$

and

$$(1 - x^2)\, y'' - 3x\, y' + n(n+2)\, y = 0,$$

for the polynomials of the first and second kind, respectively.

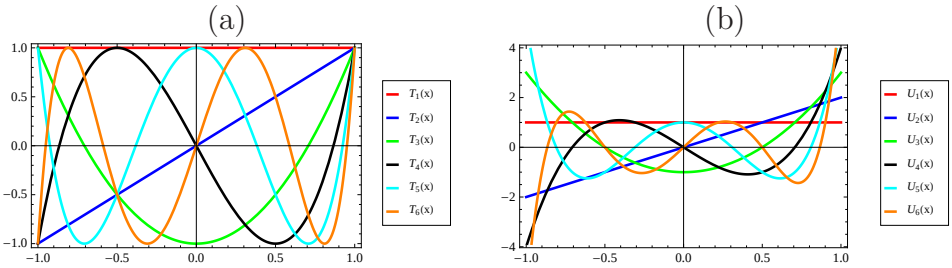


FIG. 1.1: The first six Chebyshev polynomials of the first kind $T_n(x)$ (a) and of the second kind $U_n(x)$ (b) on the interval $x \in [-1, 1]$.

The Chebyshev polynomials of the first kind are defined by the recurrence relation

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_{n+1}(x) &= 2x\,T_n(x) - T_{n-1}(x). \end{aligned}$$

The first few Chebyshev polynomials of the first kind are (see Fig. 1.2.3 (a))

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_2(x) &= 2x^2 - 1, \\ T_3(x) &= 4x^3 - 3x, \\ T_4(x) &= 8x^4 - 8x^2 + 1, \\ T_5(x) &= 16x^5 - 20x^3 + 5x. \end{aligned}$$

Notice, that $|T_n| \leq 1$. Since T_n is a degree n polynomial we can observe as expected that it has n zeros, which in this case are real and distinct and located in the interval $[-1, 1]$,

$$x_j = \cos\left(\frac{\pi(2j-1)}{2n}\right), \qquad j = 1, \ldots, n. \tag{1.17}$$

The zeros are also known as the *Chebyshev-Gauss (CG) points*.
The recurrence relation for the Chebyshev polynomials of the second kind U_n reads

$$\begin{aligned} U_0(x) &= 1, \\ U_1(x) &= 2x, \\ U_{n+1}(x) &= 2x\,U_n(x) - U_{n-1}(x). \end{aligned}$$

The first few Chebyshev polynomials of the second kind are (see Fig. 1.2.3 (b))

$$\begin{aligned} U_0(x) &= 1, \\ U_1(x) &= 2x, \\ U_2(x) &= 4x^2 - 1, \\ U_3(x) &= 8x^3 - 4x, \\ U_4(x) &= 16x^4 - 12x^2 + 1, \\ U_5(x) &= 32x^5 - 32x^3 + 6x. \end{aligned}$$

The n roots of U_n can be found as

$$x_j = \cos\left(\frac{\pi j}{n+1}\right), \qquad j = 1, \ldots, n.$$

Intervals $y \in [a,\, b]$ other than $x \in [-1,\, 1]$ are easily handled by the change of variables

$$y = \frac{x - \frac{1}{2}(b+a)}{\frac{1}{2}(b-a)} \Leftrightarrow x = \frac{(b-a)y + (b+a)}{2}.$$

Orthogonality

Both the $\{T_n\}_{n=0}^{\infty}$ and the $\{U_n\}_{n=0}^{\infty}$ form a sequence of orthogonal polynomials. The polynomials of the first kind are orthogonal with respect to the weight $\frac{1}{\sqrt{1-x^2}}$ on the interval $[-1, 1]$, i.e., one can write:

$$\int_{-1}^{1} T_n(x) T_m(x) \frac{dx}{\sqrt{1-x^2}} = \begin{cases} 0, & n \neq m, \\ \pi, & n = m = 0, \\ \pi/2, & n = m \neq 0 \end{cases}$$

and the polynomials of the second kind are orthogonal with respect to the weight $\sqrt{1-x^2}$, i.e.,

$$\int_{-1}^{1} U_n(x) U_m(x) \sqrt{1-x^2} dx = \begin{cases} 0, & n \neq m, \\ \pi/2, & n = m. \end{cases}$$

Trigonometric definition

The Chebyshev polynomials of the first kind are defined through the identity

$$\boxed{T_n(\cos\theta) = \cos(n\theta),} \tag{1.18}$$

or, equivalently,

$$T_n(x) = \cos(n \arccos x), \text{ for } \quad n \in \mathbb{N}_0, \tag{1.19}$$

whereas the Chebyshev polynomials of the second kind satisfy:

$$U_n(\cos(\theta)) = \frac{\sin((n+1)\theta)}{\sin\theta}, \text{ for } \quad n \in \mathbb{N}_0. \tag{1.20}$$

Chebyshev series and discrete transformations

The set of Chebyshev polynomials form a complete basis set in the appropriate space, so that a function $u(x)$, $x \in [-1, 1]$ in the same space can be expressed via the expansion:

$$u(x) = \sum_{n=0}^{\infty} \widehat{u}_n T_n(x).$$

This sum is called to be *a Chebyshev series* or *a Chebyshev expansion*. Since the Chebyshev polynomials form an orthogonal basis, the coefficients $\widehat{u}_n$ can be determined easily through the application of an inner product,

$$\widehat{u}_n = \frac{2}{\pi\, c_n} \int_{-1}^{1} \frac{u(x) T_n(x)}{\sqrt{1-x^2}} dx,$$

where

$$c_n = \begin{cases} 2, & n = 0, \\ 1, & n \geq 1. \end{cases} \tag{1.21}$$

Now if we define a new function $\widetilde{u}$ by $\widetilde{u}(\theta) = u(\cos\theta)$, then the expansion above can be rewritten as

$$\widetilde{u}(\theta) = \sum_{n=0}^{\infty} \widehat{u}_n \cos n\theta.$$

Hence, the Chebyshev series for u corresponds to a cosine series for $\widetilde{u}$.
Now we can also introduce a discrete grid $\{x_j\}_{j=0}^{N}$ consisting of $N+1$ points. The actual numerical representation of a function u is then the polynomial formed from the *discrete coefficients*, depending on the actual grid. Notice that several possible grids exist. However, the it is well known that polynomial interpolation in equally spaced points can be troublesome because of *Runge's phenomenon*. Runge's phenomenon is a problem that occurs when using polynomial interpolation with polynomials of high degree. The problem

can be minimized by using nodes that are distributed more densely towards the edges of the interva. A standard example of such a set of nodes is Chebyshev nodes (1.17), for which the maximum error is guaranteed to diminish with increasing polynomial order. Another example is *the Chebyshev-Gauss-Lobatto (CGL) points*

$$x_j = \cos\left(\frac{j\pi}{N}\right), \qquad j = 0, \ldots, N. \tag{1.22}$$

The CGL points consist of $n-1$ extrema of T_n plus the endpoints of the interval $[-1, 1]$. For the grid, defined by CG (1.17) and for CGL nodes (1.22) respectively , the discrete approximation read:

- *CG nodes*: $x_k = -\cos\left(\frac{\pi(2k+1)}{2(N+1)}\right), \quad k = 0, \ldots, N$

  $$u = \sum_{n=0}^{N} {}''\widehat{u}_n T_n(x),$$

  where the notation $''$ on the summation means that the first and the last terms are divided by $1/2$ and

  $$\widehat{u}_k = \frac{2}{N} \sum_{k=0}^{N} {}''u(x_k) T_n(x_k).$$

- *CGL nodes*: $x_k = -\cos\left(\frac{k\pi}{N}\right), \quad k = 0, \ldots, N$

  $$u = \sum_{n=0}^{N} {}'\widehat{u}_n T_n(x),$$

  where the notation $'$ means, that the first term in this sum is divided by $1/2$ and

  $$\widehat{u}_k = \frac{2}{N+1} \sum_{k=0}^{N} u(x_k) T_n(x_k).$$

Differentiation

By differentiating the polynomials in their trigonometric forms, one can obtain:

$$\begin{aligned} \frac{d}{dx} T_n(x) &= nU_{n-1} = \frac{n \sin n\theta}{\sin \theta}, \\ \frac{d^2}{dx^2} T_n(x) &= n\frac{(n+1)T_n - U_n}{x^2 - 1} = -\frac{n^2 \cos n\theta}{\sin^2 \theta} + \frac{n \cos\theta \, \sin n\theta}{\sin^3 \theta}. \end{aligned}$$

Additionaly, it can be shown that:

$$\begin{aligned} \left.\frac{d^2T_n}{dx^2}\right|_{x=1} &= \frac{n^4-n^2}{3}, \\ \left.\frac{d^2T_n}{dx^2}\right|_{x=-1} &= (-1)^n\frac{n^4-n^2}{3}. \end{aligned}$$

The recurrence relationship of the derivative of Chebyshev polynomials can also be derived:

$$2T_n(x) = \frac{1}{n+1}\,\frac{d}{dx}T_{n+1}(x) - \frac{1}{n-1}\,\frac{d}{dx}T_{n-1}(x), \qquad n \in \mathbb{N}. \tag{1.23}$$

Differentiation recurrence relations

Let us compute , e.g., the m's derivative from the function $u(x)$. Formally one can write the expansion

$$\frac{d^m u}{dx^m} = \sum_{k=0}^{N} a_k^{(m)} T_k(x),$$

where the subscript (m) denotes the unknown coefficients of the m'th derivative. Using relation (1.23), they can be computed from the corresponding coefficients of the $m-1$'th derivative by the recurrence relation (in decreasing order) [5]:

$$a_N^{(m)} = a_{N-1}^{(m)} = 0, \tag{1.24}$$

$$a_{k-1}^{(m)} = \frac{1}{c_{k-1}}\left(2k\,a_k^{(m-1)} + a_{k+1}^{(m)}\right), \quad k = N-1,\, N-2,\, \ldots,\, 1, \tag{1.25}$$

and c_n are given in (1.21)

1.2.4 An Example: A Heat Equation

Consider a one-dimensional heat equation of the form

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \tag{1.26}$$

$u = u(x,t)$ on the interval $x \in [-1,\, 1]$ with initial condition

$$u(x,\, 0) = u_0(x), \qquad \forall x \in [-1,\, 1] \tag{1.27}$$

and Dirichlet boundary conditions

$$u(-1,\, t) = u(1,\, t) = 0, \qquad \forall t > 0. \tag{1.28}$$

Let us seek for the solution of (1.26) by means of *the Chebyshev collocation method.*, so the solution $\widetilde{u}(x,t)$ of the (1.26) can be written as

$$\widetilde{u}(x,t) = \sum_{k=0}^{N} \widehat{u}_k(t) T_k(x), \quad k = 0,\, 1,\, \ldots,\, N.$$

According to the collocation method the test functions are the shifted Dirac delta-functions

$$\chi_j(x) = \delta(x - x_j), \qquad j = 1,\, \ldots,\, N-1,$$

where x_j are collocation points in $[-1,\, 1]$, e.g., CG or CGL points (1.17,1.22). Then the condition for the residual is equivalent to

$$\partial_t \widetilde{u}(x_j, t) = \partial_{xx} \widetilde{u}(x_j, t)$$

with boundary conditions

$$\widetilde{u}(-1, t) = \widetilde{u}(1, t) = 0$$

and initial condition

$$\widetilde{u}(x_k,\, 0) = u_0(x_k), \quad k = 0,\, \ldots,\, N.$$

A particulary choice of the nodes

$$x_j = \cos\left(\frac{\pi j}{N}\right)$$

and the recurrence relation (1.24) lead to the following expression ($T_k(x_j) = \cos(\pi jk/N)$):

$$\partial_t \widetilde{u}(x_j,\, t) = \sum_{k=0}^{N} a_k^{(2)}(t) \cos\left(\frac{\pi jk}{N}\right).$$

1.3 The Pseudospectral Method for Nonlinear PDEs

1.3.1 Problem Statement and Solution

Up to now we have considered linear problems, which may be treated exclusively in Fourier space. We have seen that in this case spectral methods yield

a highly accurate and simple way to calculate derivatives. We now want to generalize this method to nonlinear partial differential equations (more details can be found in [6, 5]). To this end consider the PDE for the field $f(x,t)$

$$\frac{\partial}{\partial t} f(x,t) = L(f(x,t)) + N(f(x,t)) \tag{1.29}$$

The right-hand side involves a linear operator L, which may contain linear functions of f like multiplication with a constant factor or linear differential operators like the Laplacian. N denotes a nonlinear operator, which for example contains powers of f or nonlinear differential operators. The Fourier transform of equation (1.29) yields

$$\frac{\partial}{\partial t} \tilde{f}(k,t) = \tilde{L}(\tilde{f}(k,t)) + \mathcal{F}[N(f(x,t))] \tag{1.30}$$

While the Fourier transforms of the linear terms of equation (1.29) can be written down in a straight-forward manner, the nonlinear term needs further treatment. It has to be noted, that it is possible in principle to treat the nonlinear term in Fourier space, which is made clear best with an example. Consider the nonlinearity

$$N(f(x,t)) = f(x,t)f(x,t) \tag{1.31}$$

The Fourier transform is readily written down as

$$\mathcal{F}[N(f(x,t))] = \mathcal{F}(f(x,t)) * \mathcal{F}(f(x,t)) = \tilde{f}(k,t) * \tilde{f}(k,t) \tag{1.32}$$

That is, this simple quadratic nonlinearity leads to a convolution in Fourier space. Let n denote the number of grid points used for the discretization of f. Then the above expression involves n^2 operations. Of course, the situation gets even worse for a higher nonlinearity. To circumvent this problem we proceed as follows. Instead of evaluating the nonlinearity in Fourier space, we may transform $\tilde{f}(k,t)$ back to real space. The computational cost for this operation scales with $n \log n$ thanks to the *fast fourier transform* algorithm. Back in real space the multiplication may easily be done with an operation scaling with n. After evaluating the nonlinearity in real space, we transform back to Fourier space, again at a cost scaling with $n \log n$. If the nonlinearity contains not only powers of f but also linear derivatives, these may also be calculated in Fourier space and transformed back together with $\tilde{f}$. Within this procedure no operation more expensive than $n \log n$ has to be executed, yielding an excellent overall performance of the algorithm compared to the brute force way.

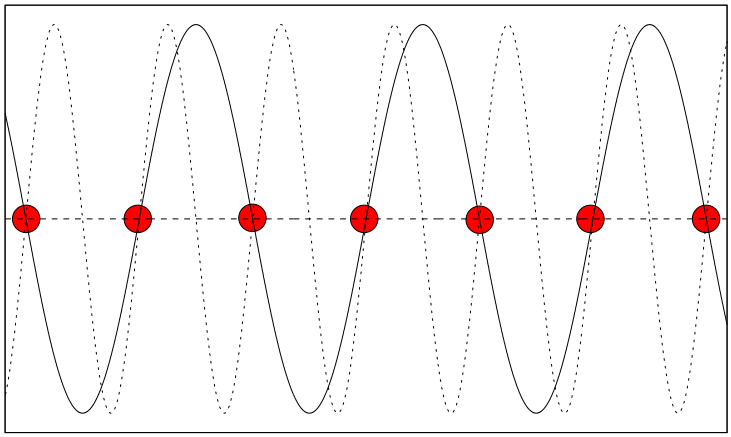


FIG. 1.2: A rapidly oscillating sinusoidal function cannot be resolved by the red grid points and hence is misinterpreted as a sinusoidal function with a lower frequency.

1.3.2 Aliasing and Dealiasing

Whenever treating a nonlinear partial differential equation, the danger of introducing aliasing errors to the numerical solution lurks around the corner. Consider for example the simple nonlinearity $f(x,t)f(x,t)$ introduced in the last paragraph. Now let f be initialized with a highly oscillating sine, which is hardly resolved by the numerical grid. Evaluating the the nonlinearity yields

$$f(x,t)f(x,t) = \sin(\omega x)\sin(\omega x) = -\frac{1}{2}\cos(2\omega x) \tag{1.33}$$

i.e., the nonlinearity introduces an even more rapid oscillating cosine, which cannot be resolved by the numerical grid. As a consequence this oscillating function is interpreted as an oscillating function with a lower frequency. This is also visualized in figure 1.2. Depending on your problem, this effect may introduce severe errors to your numerical solution or even result in a numerical instability.

There are a at least two ways which help to avoid this effect. Probably the most simple one is to choose a grid which is fine enough to accurately resolve all the scales which may appear during the course of the simulation. This is an appropriate solution for many physical systems. Considering for example model equations for pattern formation, here often a small wave number band is excited and the spectral density of the field rapidly decays far from these excited modes.

In case of, for example, Navier-Stokes turbulence the situation turns out

to be a bit different. Here, the spectral density of, say, the kinetic energy decays algebraically (with a $k^{-\frac{5}{3}}$ spectrum) indicating that a wide range of spatial modes are excited. Together with the fact, that we have a quadratic nonlinearity for this problems, a pseudospectral treatment of this problems yields an alternative solution called dealiasing. Different approaches to dealias the fields have been proposed in the literature, the easiest being the two-thirds rule introduced by Orszag. Let $k_{\max}$ denote the highest wavenumber resolved by the numerical grid. Provided, that all fields involved in calculating the nonlinearity are filtered according to

$$\tilde{f}(k,t) = \begin{cases} \tilde{f}(k,t) & \text{if} \quad k < \frac{2}{3}k_{\max} \\ 0 & \text{else} \end{cases} \tag{1.34}$$

ensures that the numerical solution is free from aliasing errors. It may be reasoned from figure 1.3, that by this truncation of the input fields in Fourier space a quadratic nonlinearity may only produce higher modes which are readily filtered after calculating the nonlinearity. It is very easy to apply this method. You only have to

- null the upper third of Fourier coefficients of all fields which are involved in calculating the nonlinearity
- null the upper third of Fourier coefficients of the array holding the Fourier coefficients of the nonlinearity after transforming back into Fourier space.

It has to be stressed that this method only helps to suppress aliasing errors and does not increase the resolution of the numerical solution. A further disadvantage of this simple method is that, being a sharp filter in Fourier space, it may introduce Gibbs oscillations to your numerical solution. So the best way to avoid aliasing errors is to accurately resolve the numerical solution of your problem not pushing it to the limit.

1.3.3 A Simple Example: Burgers Equation

Consider a real velocity field $u(x,t)$ which obeys the simple nonlinear PDE

$$\frac{\partial}{\partial t}u(x,t) + u(x,t)\frac{\partial}{\partial x}u(x,t) = \nu\frac{\partial^2}{\partial x^2}u(x,t) \tag{1.35}$$

which is known as Burgers equation. It shares the advective nonlinearity with the Navier-Stokes equation, albeit in a single spatial dimension. At

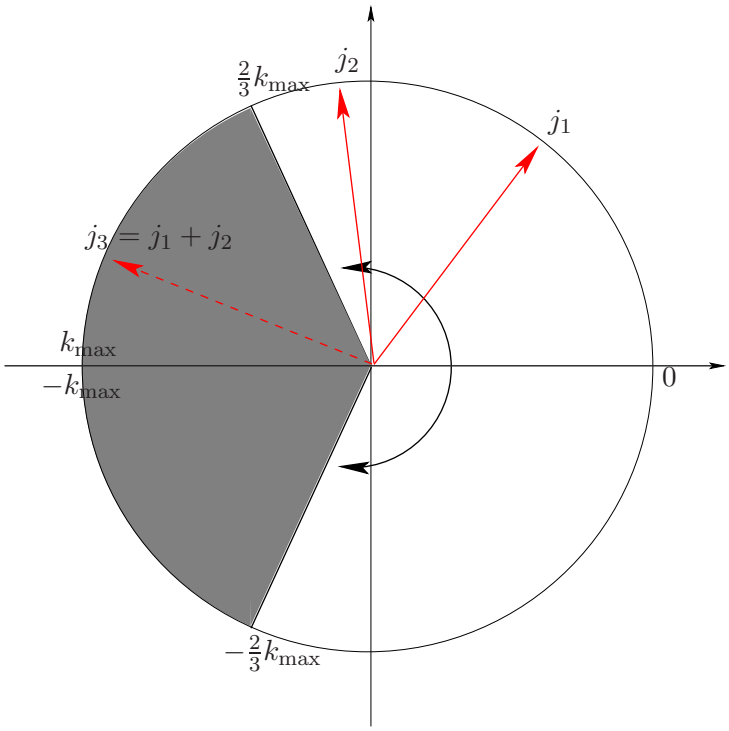


FIG. 1.3: Two-thirds rule by Orszag. A coupling of two modes corresponds to an addition of angles in this figure. After the Fourier coefficients of the input fields are nulled according to the gray-shaded areas, the sum of two arbitrary modes j_1 and j_2 results in a mode j_3 lying in the gray-shaded area. This is nulled again after the nonlinearity is transformed back to Fourier space.

the same time, the huge mathematical difficulty arising due to the nonlocal pressure term is absent in this equation. According to the above notation the linear and nonlinear operators for this PDE take the form

$$L(u(x,t)) = \nu \frac{\partial^2}{\partial x^2} u(x,t) \tag{1.36}$$
$$N(u(x,t)) = -u(x,t) \frac{\partial}{\partial x} u(x,t)$$

ν denotes the kinematic viscosity. The Burgers equation is known to steepen negative gradients leading to the formation of so-called shocks. These shocks are smoothed out by viscous effects. For a sufficiently high viscosity, this equation may efficiently be solved by a pseudospectral method, as we want to exemplify in the following. Time-stepping can be achieved with a numerical scheme of your choice, fourth-order Runge-Kutta methods turn out to do a rather good job. After initializing an initial condition $u(x, t = 0)$ the field is transformed to Fourier space yielding $\tilde{u}(k, t = 0)$. The right hand side then may be treated in several easy steps according to

- dealias $\tilde{u}(k,t)$
- calculate the derivative according to $ik\tilde{u}(k,t)$
- transform $\tilde{u}(k,t)$ and $ik\tilde{u}(k,t)$ back to real space
- calculate the nonlinearity $N(x,t) = u(x,t) \cdot \frac{\partial}{\partial x} u(x,t)$
- transform $N(x,t)$ back to Fourier space
- dealias $\tilde{N}(k,t)$
- calculate the Laplacian $-k^2\tilde{u}(k,t)$
- build the output array according to $\tilde{N}(k,t) - \nu k^2 \tilde{u}(k,t)$

Taking for example a sinusoidal function as an initial conditions, negative gradients steepen yielding the characteristic shock structure.

Chapter 2

Model Equations

2.1 The Ginzburg-Landau Equation

Ginzburg-Landau equations have been used to model a wide variety of physical systems (see, e.g., [1]). In the context of pattern formation the Real Ginzburg-Landau Equation (RGLE) was first derived as long-wave *amplitude equation* in the context of convection in binary mixtures near the onset of instability [16], [19]. The Complex Ginzburg-Landau Equation (CGLE) was first derived in the studies of Poiseuille flow [21] and reaction-diffusion systems [10].
Let us consider the conditions under which the real and complex Ginzburg-Landau equations arise. For simplicity we restrict attention to one spatial dimension. However, the results can be easily generalised to two- and three-dimensional cases.

2.1.1 The Real Ginzburg-Landau Equation

Let us consider a system

$$\partial_t u = \mathcal{L}(\sigma)u, \qquad u = u(x,t) \tag{2.1}$$

with a nonlinear operator $\mathcal{L}$, depending on some control parameter σ. Suppose that the system (2.1) admits a homogeneous solution $u = u_0$ and (2.1) undergoes a finite-wavelength instability as σ is varied, e.g., becomes positive. That is, if we consider evolution of the Fourier mode $\exp(ikx + \lambda t)$ the growthrate $\mathrm{Re}(\lambda)$ behaves as follows: for $\sigma < 0$ all modes are decaying ($\mathrm{Re}(\lambda) < 0$) and u_0 is stable. For $\sigma = 0$, a critical wavenumber k_c gains neutral stability and for $\sigma > 0$ there is a narrow band of wavenumbers around k_c

where the growthrate $\mathrm{Re}(\lambda)$ is positive. Let us also assume that the instability we are interested in is *supercritical*, i.e., the nonlinearities saturate so that the resulting patterns above the threshold ($\sigma \ll 1$) have small amplitude and a wavelength close to $2\pi/k_c$.

If $\mathrm{Im}(\lambda) = 0$ the unstable modes are growing in time for positive values of σ but each mode is stationary in space. Thus, close to theshold, the dynamics of (2.1) can be written as

$$u = u_0 + A(x,t)e^{ik_c x} + \overline{A}(x,t)e^{-ik_c x} + h.o.t.,$$

where $A(x,t)$ denotes the complex amplitude. Then, to lowest order in σ, and after rescaling, the amplitude A obeys *the real Ginzburg-Landau equation (RGLE)*:

$$\frac{\partial A}{\partial t} = \frac{\partial^2 A}{\partial x^2} + \sigma A - |A|^2 A. \tag{2.2}$$

With an additional rescaling

$$x \mapsto \sigma^{-1/2}x,\ t \mapsto \sigma^{-1}x,\ A \mapsto \sigma^{1/2}A,$$

the contral parameter σ can be scaled out from Eq. (2.2), i.e.,

$$\boxed{\frac{\partial A}{\partial t} = \frac{\partial^2 A}{\partial x^2} + A - |A|^2 A.} \tag{2.3}$$

Notice that Eq. (2.3) arises naturally near any stationary supercritical bifurcation if the system (2.1) features translational invariance and is reflection symmetric ($x \mapsto -x$). Translational invariance, e.g., implies that (2.3) has to be invariant under $A \mapsto Ae^{i\phi}$. Notice also that Eq. (2.3) can be rewritten in the form

$$\frac{\partial A}{\partial t} = -\frac{\delta \mathcal{F}}{\delta A^*}, \quad \mathcal{F} = \int dx \left(|\frac{\partial A}{\partial x}|^2 - \varepsilon |A|^2 + \frac{1}{2}|A|^4 \right),$$

and thus, $\mathcal{F}$ plays role of a Lyapunov functional. The next point to emphasize is that Eq. (2.3) has a stationary *phase winding* solution

$$A = a_0 e^{iqx}, \qquad q^2 = 1 - a_0^2$$

describing steady state periodic patterns.

2.1.2 The Complex Ginzburg-Landau Equation

Now let us consider the case $\mathrm{Im}(\lambda) := \omega_c \neq 0$, so each mode corresponds to a travelling wave. In this case we can write the solution of (2.1) as

$$u = u_0 + A(x,t)e^{ik_c x + i\omega_c t} + \overline{A}(x,t)e^{-ik_c x - i\omega_c t} + h.o.t.,$$

and, after rescaling, the equation for the complex amplitude A reads

$$\boxed{\frac{\partial A}{\partial t} = (1+i\alpha)\frac{\partial^2 A}{\partial x^2} + A - (1+i\beta)|A|^2 A}, \tag{2.4}$$

where α and β are parameters. This equation is referred to as the complex Ginzburg-Landau Equation (CGLE). Notice that the RGLE (2.3) is simply a special case of the CGLE (2.4) with $\alpha = \beta = 0$. Notice also, that in the limit case α, $\beta \to \infty$ Eq. (2.4) reduces to the Nonlinear Schrödinger Equation, which has, for instance, well-known soliton solutions.

Plane waves and their stability

The simplest solutions of CGLE are plane wave solutions, which take the form

$$A = a_0 e^{iqx + i\omega t}, \tag{2.5}$$

where

$$a_0^2 = 1 - q^2, \qquad \omega = -\beta - (\alpha - \beta)q^2.$$

The expression for ω illustrates that the coefficient α and β mesure linear and nonlinear dispersion (the dependence of the frequency of the waves on the wavenumber), respectively. In order to investigate the stability of (2.5) we seek the solution in the form

$$A = \left(a_0 + \widetilde{a}_+ e^{ikx + \lambda t} + \widetilde{a}_- e^{-ikx + \lambda^* t}\right) e^{iqx + i\omega t},$$

where $\widetilde{a}_\pm$ denote the amplitudes of the small peturbations. After substitution of this expression into (2.4) one can find an equation for the growth rate λ. By expanding this equation for small k (the long-wavelength limit) one obtains [1]

$$\lambda = -2iq(\alpha - \beta)k - \left[1 + \alpha\beta - \frac{2q^2(1+\beta^2)}{a_0^2}\right]k^2 + \mathcal{O}(k^3). \tag{2.6}$$

Thus, travelling waves solutions are long-wave stable as long as the condition

$$1 + \alpha\beta - \frac{2q^2(1+\beta^2)}{a_0^2} > 0$$

holds. That is to say that one has a stable range of wave vectors with

$$q^2 < \frac{1+\alpha\beta}{2\beta^2 + \alpha\beta + 3}$$

including the band centre ($q = 0$) state as long as the so-called *Benjamin-Feir-Newell criterion*

$$1 + \alpha\beta > 0 \tag{2.7}$$

holds. Notice that the Benjamin-Feir instability criterion is a generalisation of *the Eckhaus instability*. For example, for $\alpha = \beta$, the stability condition reduces to the well-known Eckhaus condition $q^2 < 1/3$. The next point to emphasize is that for $\alpha \neq \beta$ and $q \neq 0$ the destabilizing modes have a group velocity $v_g = 2q(\alpha - \beta)$, i.e., the instability is of a convective nature.

2.1.3 Plane waves: Numerical Treatment

Our goal is to solve Eq. (2.4) by means of pseudospectral method and ETD2 exponential time-stepping (52) (see Appendix (.5)). According to the notations, introduced in Appendix (.5), Eq. (2.4) in the Fourier space becomes

$$\frac{d\widehat{A}}{dt} = \underbrace{(1 - k^2(1+i\alpha))}_{q}\,\widehat{A} - \underbrace{\mathcal{F}[(1+i\beta)|A|^2 A]}_{\mathcal{N}}, \tag{2.8}$$

i.e., scheme (52) can be applied:

$$\widehat{A}_{n+1} = \widehat{A}_n e^{qh} + \mathcal{N}_n \frac{(1+hq)e^{qh} - 1 - 2hq}{hq^2} + \mathcal{N}_{n-1}\frac{-e^{qh} + 1 + hq}{hq^2}. \tag{2.9}$$

We start from the simulation of the plane waves. Let us choose parameters α and β such that there exists a stable range of wavenumbers and then simulate the CGLE with an initial condition of small noise of order 0.01 about $A = 0$. Other parameters are

`Constants`	$(\alpha,\ \beta) = (1, 2)$
`Domain size`	L=100
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$

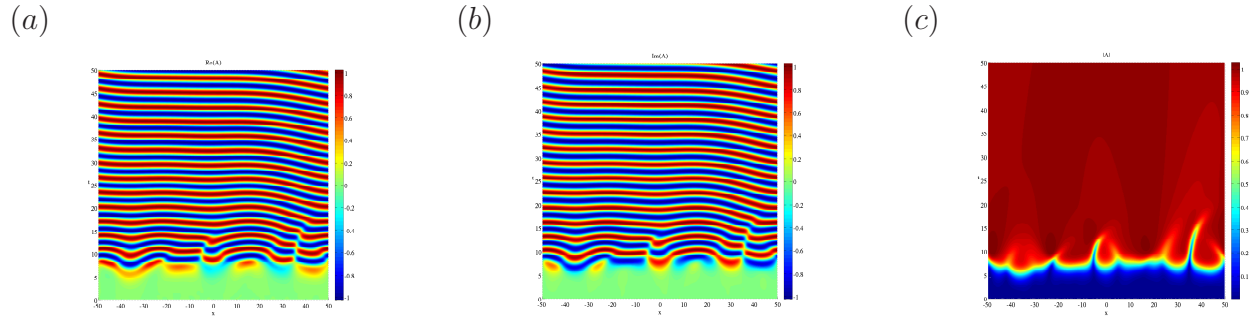


FIG. 2.1: Space-time plots of (a) Re(A), (b) Im(A) and (c) $|A|$ for the case $(\alpha, \beta) = (1, 2)$.

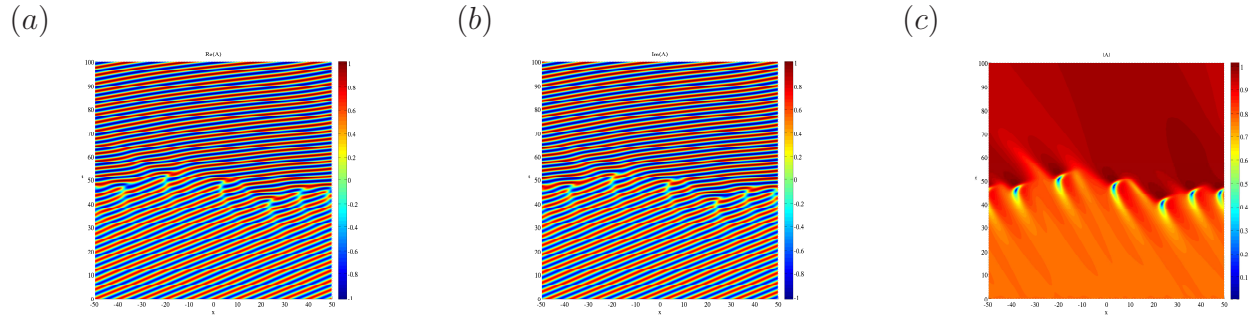


FIG. 2.2: Space-time plots of (a) Re(A), (b) Im(A) and (c) $|A|$ in the case of the Benjamin-Feir instability.

The selection of a plane wave can be seen on Fig. (2.1.3). One can see that $|A|$ quickly converges to a non-zero constant value.
In order to simulate the Benjamin-Feir instability let us consider the same parameter space but using a linearly unstable plane wave as an initial condition:

`Constants`	$(\alpha, \beta) = (1, 2)$
`Domain size`	L=100
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x,0) = \sqrt{1 - \left(\frac{20\pi}{L}\right)^2}\exp(i\frac{20\pi}{L}x) + \text{noise}$

The result is presented on Fig. (2.1.3) In can be seen that a new plane wave is selected with wavenumber lying inside the band of stability. The process of selecting the new plane wave gives rise to *'defects' or phase singularities* (points where $A = 0$).

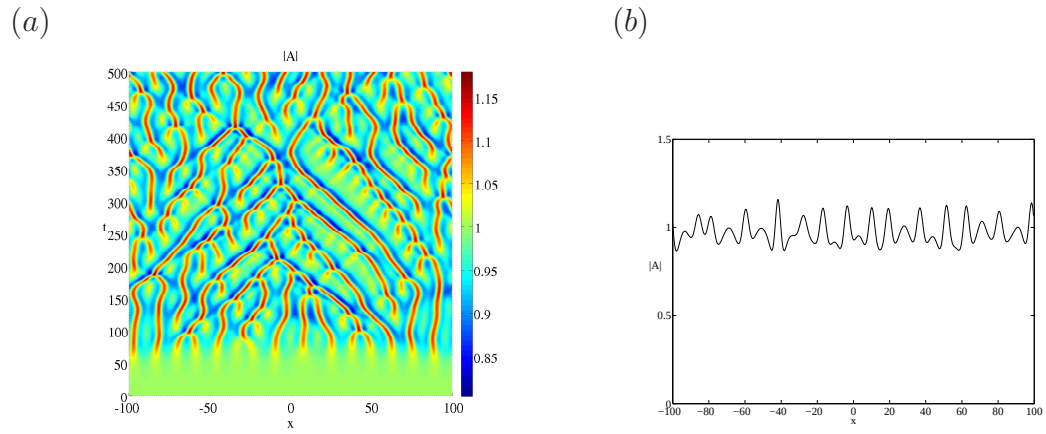


FIG. 2.3: Phase turbulence regime observed for $(\alpha,\ \beta) = (2,\ -1)$. (a) Space-time plot $|A|$. (b) The final configuration of $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha,\ \beta) = (2, -1)$
`Domain size`	L=200
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x, 0) = 1 + \text{noise}$

Spatiotemporal chaos

Now let us discuss behaviours of the solutions of the one-dimensional CGLE (2.4) when the Benjamin-Feir-Newell criterion (2.7) is violated. In this region of the parameter space $(\alpha,\ \beta)$ several different forms of spatio-temporal chaotic or disordered states have been found [2, 7]. In particular beyond the BF instability line Eq. (2.4) exhibits so-called *phase turbulence* regime (see Fig. (2.1.3)), which can be described by a phase equation of the Kuramoto-Sivashinsky type [1, 7]. As can be seen on Fig. (2.1.3 (b)), in this spato-temporally chaotic state $|A|$ never reaches zero and remains saturated, so the global phase difference becomes the constant of the motion and is conserved. Apart from phase turbulence Eq. (2.4) exhibits spatio-temporally disordered regime called *amplitude* or *defect turbulence*. The behaviour in this region is characterised by defects, where $|A| = 0$ (see Fig. (2.1.3)) Apart from spatio-temporal chaotic behaviour discribed above a so-called *bichaos* region can be found [7]. In this region, depending on the initial conditions, either defect-mediated turbulence or phase turbulence can be indicated.

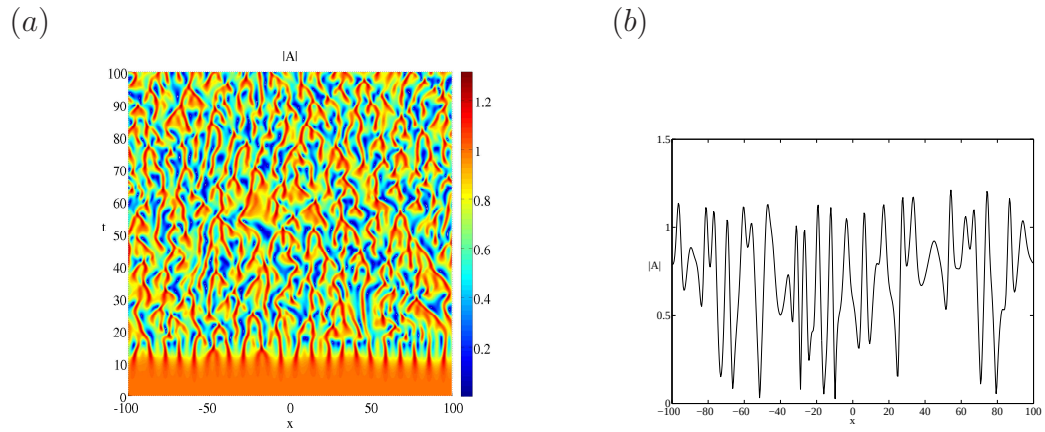


FIG. 2.4: Defect turbulence regime observed for $(\alpha,\, \beta) = (2,\, -2)$. (a) Space-time plot $|A|$. (b) The final configuration of $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha,\, \beta) = (2, -2)$
`Domain size`	$L = 200$
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x, 0) = 1 + \text{noise}$

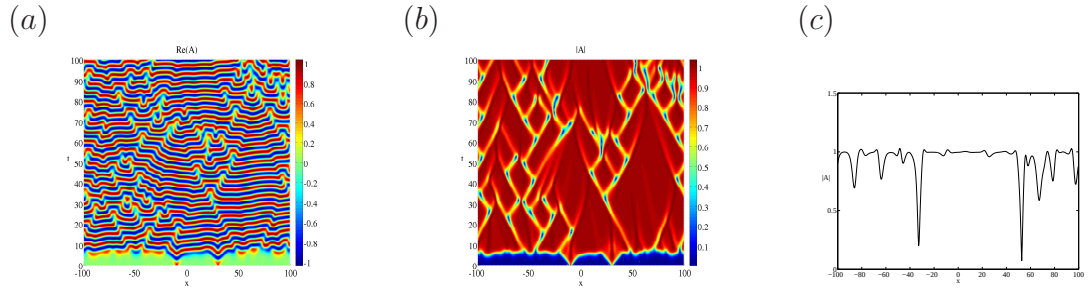


FIG. 2.5: The intermittency regime observed for $(\alpha,\ \beta) = (0.5,\ -1.5)$. (a) Space-time plot Re$(A)$; (b) Space-time plot $|A|$; (c) The final configuration of $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha,\ \beta) = (0.5, -1.5)$
`Domain size`	$L = 200$
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x,0) = \mathrm{sech}((x+10)^2) + 0.8 * \mathrm{sech}((x-30)^2) + \mathrm{noise}$

The intermittency regime

The linear stability of the plane wave solution (2.7) does not exclude the existence or coexistence of the other non-trivial solutions of (2.4). For example, the regime of spatio-temporal intermittency, where defect chaos coexists with stable plane wave was discussed in details in [7]. There, in order to avoid the stable plane wave solution, initial condition was composed of one or several localised pulses of amplitude. After a rather short transient, the typical solution consist of localized structures, separating lager regions of almost constant amplitude which are patches of stable plane wave solutions (see Fig. (2.1.3)). Figure (2.1.3) shows a more complicated intermittency scenario observed for $(\alpha,\ \beta) = (0,\ -4)$. In this case the spatial extension of the system is broken by irregular arrangements of stationary hole- and shock-like objectes separated by turbulent dynamics.

Coherent structures

Apart from plane waves, Eq. (2.4) provides a lager variety of so-called *coherent structures*. These solutions are either localized or consist of domains of regular patterns connected by localized defects or interfaces. One-

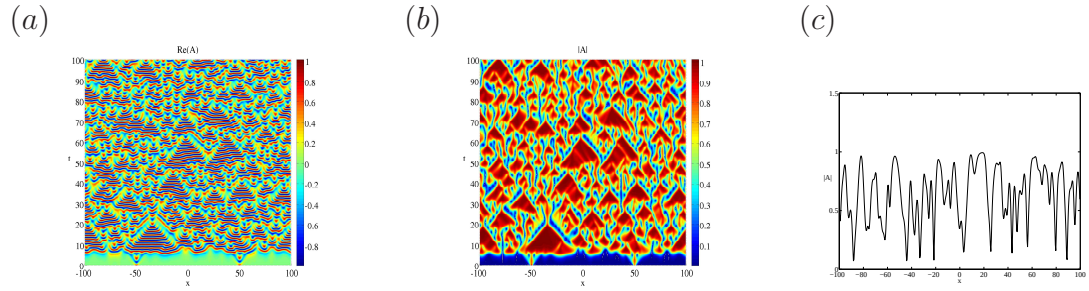


FIG. 2.6: The intermittency regime observed for $(\alpha,\ \beta) = (0,\ -4)$. (a) Space-time plot Re$(A)$; (b) Space-time plot $|A|$; (c) The final configuration of $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha,\ \beta) = (0,\ -4)$
`Domain size`	$L = 200$
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x,0) = \mathrm{sech}((x + L/4)^2) + 0.8 * \mathrm{sech}((x - L/4)^2) + \mathrm{noise}$

dimensional coherent structure can be written in the form [23]

$$A(x,t) = a(x - vt)\exp(i\phi(x - vt) - i\omega t). \tag{2.10}$$

After substitution of this equation into (2.4) lead to the system of three ordinary differential equations with respect to variables (see [23]for more details)

$$q(\xi) = \partial_\xi \phi, \quad \kappa(\xi) = \partial_\xi a/a, \quad \partial_\xi a = \partial a/\partial_\xi,$$

where $\xi = x - vt$. The resulting system of ODE's can be discussed in terms of dynamical system in pseudo-time ξ with three degrees of freedom. Accordingly to the assymptotic states the localized coherent structues can be classified as *fronts, pulses, source (holes) and sinks (shoks)*. The best known example of sink solutions are *Bekki-Nozaki holes* [3], illustrated on Fig. (2.1.3). These structures asymptotically connect plane waves of different amplitude and wavenumber and make a one-parameter family of solutions of CGLE [15]. As can be seen on Fig. (2.1.3) in the case of several holes they are separated by shoks ([1]).

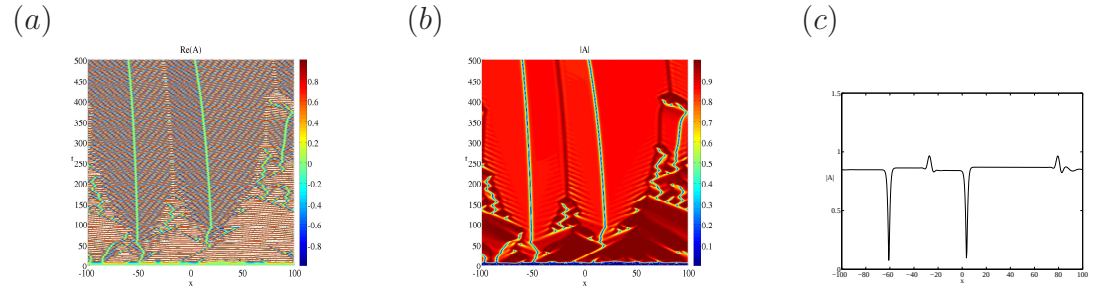


FIG. 2.7: The moving hole-shock pair observed for $(\alpha,\, \beta) = (0,\, 1.5)$. (a) Space-time plot Re$(A)$; (b) Space-time plot $|A|$; (c) The final configuration of $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha,\, \beta) = (0,\, 1.5)$
`Domain size`	$L = 200$
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 512$
`Initial condition`	$A(x, 0) =$ noise of the amplitude 0.01

2.1.4 The CGLE in 2D

The two-dimensional version of the CGLE reads

$$\frac{\partial A}{\partial t} = (1 + i\alpha)\triangle A + A - (1 + i\beta)|A|^2 A, \tag{2.11}$$

where A is a complex field. Apart of two-dimensional analogues of defect and phase turbulence Eq. (2.11) has a variety of coherent structures [1, 8]. Among others Eq. (2.11) possesses two-dimensional cellular structures known as *frosen states* (see Fig. (**??**)). They appear in the form of quasi-frosen arrangments of spiral defects surrounded by shock lines. Note that $|A|$ in this regime is stationary in time. For a complete review of the phase diagram for the two-dimensional CGLE see [8].

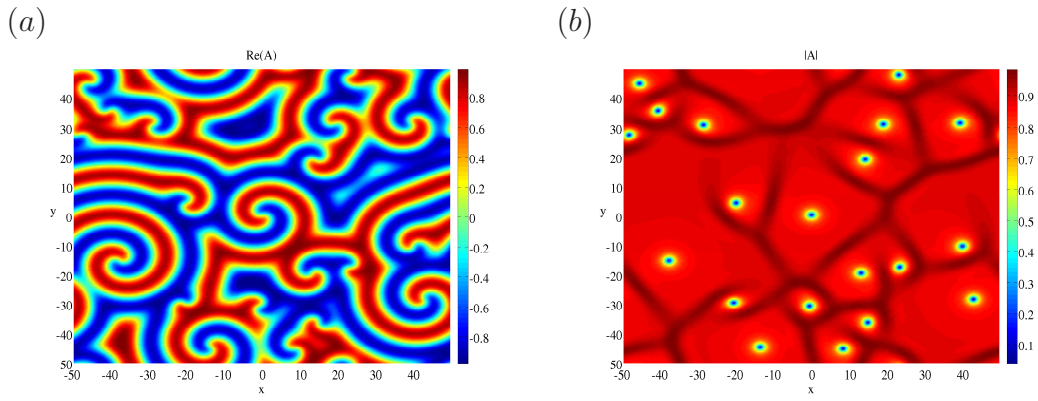


FIG. 2.8: Two-dimensional cellular structures observed for $(\alpha, \beta) = (0, 1.5)$. (a) Space-time plot Re$(A)$; (b) Space-time plot $|A|$. Other parameters: see the Table below.

`Constants`	$(\alpha, \beta) = (0, 1.5)$
`Domain size`	$L = [-100, 100] \times [-100, 100]$
`Timestep`	$h = 0.05$
`Number of grid points`	$N = 256$
`Initial condition`	$A(x, 0) =$ noise of the amplitude 0.01

Chapter 3

Numerical Simulation of Hydrodynamical Problems

In this section we will focus on partial differential equations from hydrodynamics like for example the vorticity equation in two dimensions or the Rayleigh-Bènard system. These systems can be treated effectively with pseudospectral methods, the main techniques have been introduced within the preceding paragraphs.

One of the main difficulties in simulating hydrodynamical systems lies in the nature of the physical problem. When driven far out ouf equilibrium, many hydrodynamical systems reach a turbulent state. Without giving a concise definition of this state, it may be characterized by a high degree of spatial and temporal complexity. Typical for a turbulent state is that a large number of degrees of freedom are involved, however these degrees do not fluctuate independently, but show signatures of coherence and order. Having a glimpse at a turbulent vorticity field reveals a number eddies of different sizes interacting in a complicated manner, as can be seen from the illustrations 3.1 and 3.3. These illustrations indicate, that we have to deal with a multi-scale phenomenon, where a broad band in wavenumber space is excited (see figure 3.2), a situation strongly different from the pattern forming systems of the last paragraphs. That means, a numerical simulation of such a system necessitates an accurate resolution of all temporal and spatial scales of the system. This section gives a brief overview of the numerical schemes used to get an accurate and stable solution.

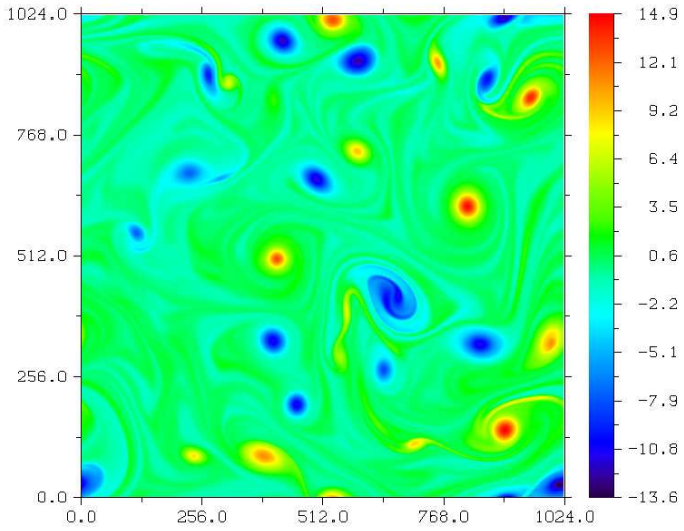


FIG. 3.1: Vorticity field from a simulation of the two-dimensional Navier-Stokes equation. The vorticity tends to organize into elliptical vortices.

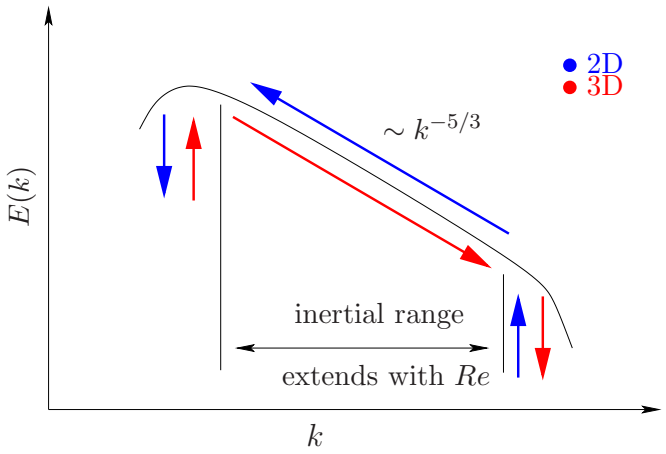


FIG. 3.2: Sketch of the energy spectrum of turbulence. In two dimensions, energy cascades to smaller wavenumbers, whereas the opposite is the case for three-dimensional turbulence. Both systems can exhibit a $k^{-5/3}$ scaling range, which is called the inertial range.

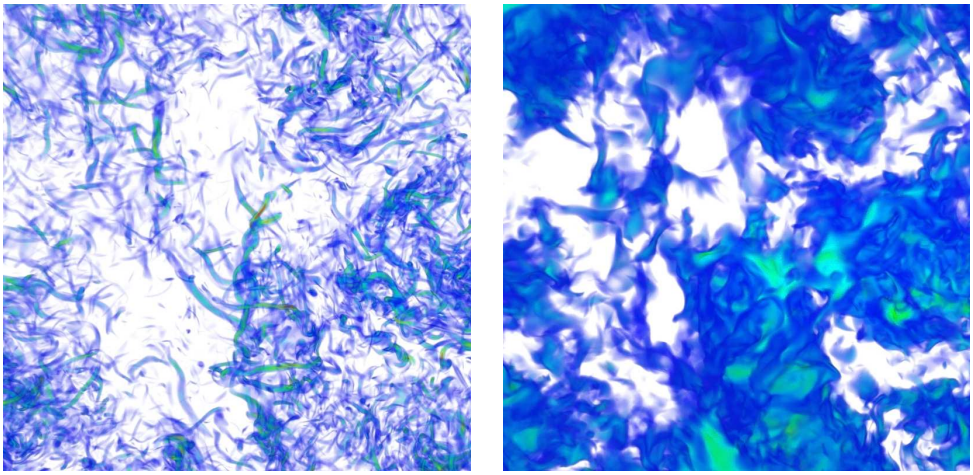


FIG. 3.3: Volume rendering of vorticity (left) and velocity (right) from a simulation of the three-dimensional Navier-Stokes equation. The vorticity tends to organize into thin filaments, which are entangled forming a complicated global structure. The velocity appears more unstructured but displays long-range correlations.

3.1 Vorticity Equation

3.1.1 Basic Equations in Real and Fourier Space

The Navier-Stokes equations read

$$\frac{\partial}{\partial t}\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + \nu\Delta\boldsymbol{u}(+\boldsymbol{F}) \tag{3.1}$$

$$\nabla\cdot\boldsymbol{u} = 0, \tag{3.2}$$

where $\boldsymbol{u}(\boldsymbol{x},t)$ denotes the velocity field in two or three dimensions. The left-hand side of this equation involves the temporal change of velocity at a fixed point in space and contribution from the nonlinear advection of the velocity field. The force acting on the fluid may be seen from the right-hand side. The fluid is driven by the negative pressure gradient, which maintains zero divergence of the velocity field in the whole domain. This term hereby is not only nonlinear, but also nonlocal, as it may be calculated as an integral over the whole fluid domain with a nonlinear integrand. Additionally the fluid diffuses with a kinematic viscosity ν and optionally is driven by external forces termed $\boldsymbol{F}$. It turns out to be very convenient to describe the fluid in terms of the curl of the velocity, the so-called vorticity $\boldsymbol{\omega} = \nabla\times\boldsymbol{u}$. By taking the curl of the Navier-Stokes equations we readily obtain the vorticity

equation

$$\frac{\partial}{\partial t}\boldsymbol{\omega} + \boldsymbol{u}\cdot\nabla\boldsymbol{\omega} = \boldsymbol{\omega}\cdot\nabla\boldsymbol{u} + \nu\Delta\boldsymbol{\omega}(+\nabla\times\boldsymbol{F}). \tag{3.3}$$

Hereby the nonlocal pressure gradient vanishes, however the nonlocality has moved to the determination of the velocity field, which is related to the vorticity via a Poisson equation

$$\Delta\boldsymbol{u} = -\nabla\times\boldsymbol{\omega}. \tag{3.4}$$

So the vorticity is advected according to a velocity field, which is determined by the spatial organization of the vorticity in the whole fluid domain. A second important dynamical influence is the vortex stretching term on the right-hand side of the equation, which allows for a local self-amplification of vorticity in three dimensions, leading to thin tornado-like filaments as depicted in figure 3.3.

In the following we will focus on two-dimensional fluid motion. In this case the vorticity is described by a (pseudo-)scalar $\omega(x,y) = \omega(x,y)\boldsymbol{e}_z$ and a two-dimensional velocity field $\boldsymbol{u}(x,y)$. As an immediate consequence, the vortex stretching term vanishes, leading to

$$\frac{\partial}{\partial t}\omega = -\boldsymbol{u}\cdot\nabla\omega + \nu\Delta\omega, \tag{3.5}$$

i.e. the only nonlinearity in the equation is the advection of the vorticity field. For an efficient numerical solution of this equation it turns out to be useful to rewrite the nonlinear term according to

$$\nabla\times(\boldsymbol{u}\times\boldsymbol{\omega}) = \underbrace{\boldsymbol{u}(\nabla\cdot\boldsymbol{\omega})}_{=0} - \underbrace{\boldsymbol{\omega}(\nabla\cdot\boldsymbol{u})}_{=0} + \underbrace{\boldsymbol{\omega}\cdot\nabla\boldsymbol{u}}_{=0} - \boldsymbol{u}\cdot\nabla\boldsymbol{\omega}, \tag{3.6}$$

where the first two terms vanish as both the velocity and vorticity field are divergence free, and the third term is just the vortex stretching term, which also vanishes in two dimensions. In Fourier space this equations reads

$$\frac{\partial}{\partial t}\tilde{\omega} = i\boldsymbol{k}\times\mathcal{F}[\boldsymbol{u}\times\boldsymbol{\omega}] - \nu k^2\tilde{\omega}, \tag{3.7}$$

such that the nonlinearity which has to be treated real space is just the cross product of the two-dimensional velocity and the one-dimensional vorticity, $\boldsymbol{\omega} = \omega\boldsymbol{e}_z$. In Fourier space equation (3.4) can be inverted easily, allowing for a local computation of the Fourier coefficients of the velocity field according to

$$\tilde{\boldsymbol{u}} = \frac{i\boldsymbol{k}\times\tilde{\boldsymbol{\omega}}}{k^2}, \tag{3.8}$$

such that the two velocity components are given according to

$$\tilde{u}_x = \frac{i\, k_y\, \tilde{\omega}}{k^2} \qquad \text{and} \qquad \tilde{u}_y = -\frac{i\, k_x\, \tilde{\omega}}{k^2}. \tag{3.9}$$

3.2 Some Notes on the Numerical Stability

3.2.1 Time stepping

For the numerical simulation of the Navier-Stokes equation a number of different time-stepping schemes may be used. From a very stable and accurate fourth order Runge-Kutta method to a fast and memory saving second-order Adams-Bashforth scheme everything is possible and can be adjusted to your personal demands on accuracy, computational efficiency and availability of memory. The linear term is often treated implicitly or with integrating factor techniques. However, in the turbulent state the main constraint regarding stability arises from the advective term, such that the improved stability of these techniques are of minor importance in this context.

3.2.2 Aliasing and Spatial Resolution

As mentioned in the introduction turbulence involves the excitation of a broad wave number band, which makes hydrodynamical simulations vulnerable for aliasing error. Luckily we have a quadratic nonlinearity, which may be de-aliased with Orszag's two-thirds rule, which has been introduced in the context of the Burgers equation. The nonlinearity $\tilde{\boldsymbol{n}} = \mathcal{F}[\boldsymbol{u} \times \boldsymbol{\omega}]$ has to be treated according to

$$\tilde{\boldsymbol{n}}(\boldsymbol{k}, t) = \begin{cases} \tilde{\boldsymbol{n}}(\boldsymbol{k}, t) & \text{if} \quad k < \frac{2}{3} k_{\max} \\ 0 & \text{else} \end{cases} \tag{3.10}$$

When our simulation domain is $[0 : 2\pi] \times [0 : 2\pi]$ and resolved by N grid points, we have $k_{\max} = \frac{N}{2}$. In general it is important to accurately resolve the spatial structure of the flow. This may be checked by visual inspection of the vorticity field and by monitoring the energy spectrum in Fourier space, which may develop an unphysical bottleneck effect and a steepening of the highest modes, if the flow is not accurately resolved.

3.2.3 The Courant-Friedrichs-Levy Condition

Without giving rigorous proof, we will now introduce the Courant-Friedrichs-Levy condition for the numerical stability of the time-stepping scheme. The

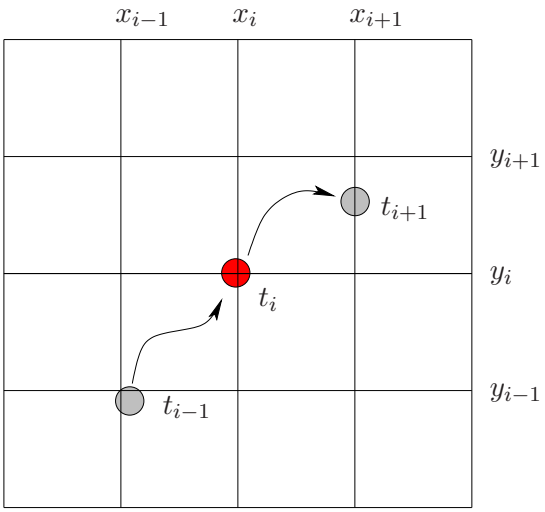


FIG. 3.4: Schematic sketch for the CFL condition. A Lagrangian particle is located at the position (x_i, y_i) at time t_i. To guarantee numerical stability, the particle may not travel more than one grid cell per time step. Hence the time step is chosen a bit too large in this figure.

basic idea here is, that information (e.g. in form of a fictive fluid particle) may not travel more than one grid cell per time step. In hydrodynamical systems two different mechanisms are present: diffusion and advection, each of them giving an independent constraint on the time step.

To evaluate the constraint of the advective term, we consider the maximal velocity in the field

$$u_{\max} = \max\{|u_x|, |u_y|\}. \tag{3.11}$$

The grid spacing is $\Delta x = \frac{L}{N}$, where L is the physical length of the simulation domain (typically 2π) and N is the number of grid points in one direction. Traveling less than one grid cell per time step now just means $u_{\max}\Delta t_{\text{adv}} < \Delta x$ yielding

$$\Delta t_{\text{adv}} = \frac{\lambda L}{u_{\max} N} \qquad \text{with} \qquad \lambda < 1. \tag{3.12}$$

λ is the CFL number, which is a $\mathcal{O}(1)$ constant depending on the time marching scheme. Usually λ is chosen less than one. For the diffusive constraint a similar reasoning applies. The diffusion law for the mean square displacement reads in two dimensions

$$\langle x^2 \rangle = 4\nu t, \tag{3.13}$$

such that traveling less than one grid cell per time step means $4\nu\Delta t_{\text{diff}} < \Delta x^2$, yielding

$$\Delta t_{\text{diff}} = \frac{\lambda L^2}{4\nu N^2} \qquad \text{with} \qquad \lambda < 1. \tag{3.14}$$

For explicit time stepping schemes we have to choose

$$\Delta t = \min\{\Delta t_{\text{adv}}, \Delta t_{\text{diff}}\} \tag{3.15}$$

in order to obtain a stable simulation. For implicit time stepping schemes or integrating factor techniques the diffusive restriction is redundant. It is important to note, that with these conditions a very easy implementation of adaptive time stepping is at hand, by evaluating the minimal allowed Δt each time step.

3.3 Implementation of Boundary Conditions

3.3.1 The Method

Up to now we have mostly considered periodic boundary conditions. In most real world flows and experimental setups boundaries play an important role. In some situtations they may be regarded more or less as experimental imperfections causing deviations from the ideal situation (at least when it comes to asking a theoretician). However, more often they play a decisive role, giving rise to a number of physical effects. Hydrodynamical systems are a good example for the latter case, for example boundary layers of turbulent fluids are an intensively studied subject, important both from theoretical and application side.

The numerical implementation of boundaries depends on the numerical scheme used. While (simple) boundaries may be implemented in finite differences and Chebychev schemes in a straight forward manner, the implicit periodicity of pseudospectral schemes seem to be a problem. However, there is a simple and efficient way to impose even complex boundary conditions on pseudospectral schemes, called volume or Brinkmann penalization. These methods have been studied extensively in recent years (see e.g. [14, 18] and references therein), and have been benchmarked in a number of situations against more traditional numerical schemes.

The main idea is to model solid objects in the flow domain as a porous medium with permeability ϵ. The locations of these objects are defined by a

mask function according to

$$H(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \text{ is within the object,} \\ 0 & \text{else.} \end{cases} \tag{3.16}$$

With the help of this mask a so-called Darcy drag term is added to the Navier-Stokes equations according to

$$\frac{\partial}{\partial t}\boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \Delta \boldsymbol{u} - \frac{1}{\epsilon} H \boldsymbol{u}. \tag{3.17}$$

This term adds a drag to the fluid within the penalized volume and at its boundaries [14]. Recall that this method is very similar to the ramp method used e.g. in case of the Swift-Hohenberg equation. However, here we have to deal with a vectorial PDE. It may be shown mathematically that within the limit of vanishing permeability no-slip boundary conditions are imposed at the boundaries of the object, i.e.

$$u_{\parallel} = u_{\perp} = 0 \quad \text{at } \delta V \tag{3.18}$$

As in the vicinity of such a boundary steep velocity gradients may occur, the problem becomes numerically stiff. In addition to the usual CFL constraints, the permeability introduces an additional constraint on the time step Δt, which has to be less or equal ϵ. Whenever this is a severe problem regarding computational ressources, an implicit treatment of this term is necessary. Additionally integrating factor techniques have proven to yield good results in this context.

3.3.2 Example: Dipole-Wall Collision

A benchmark case studied in the literature is the so-called dipole-wall collision ([14] and references therein). Here, a vortex dipole consisting of two colliding Lamb-Oseen vortices collides with a solid wall. When approaching the obstacle, a boundary layer of strong thin vorticity develops and subsequently detaches from the walls, giving birth to new vortices. This example shows, that boundaries have strong influences on the flow topology.

3.3.3 Passive Scalar Advection

Passive Scalar advection plays an important role in many natural and industrial environments. The list of examples is merely endless, but the salt and temperature concentration in the oceans or spreading of pollutants in the

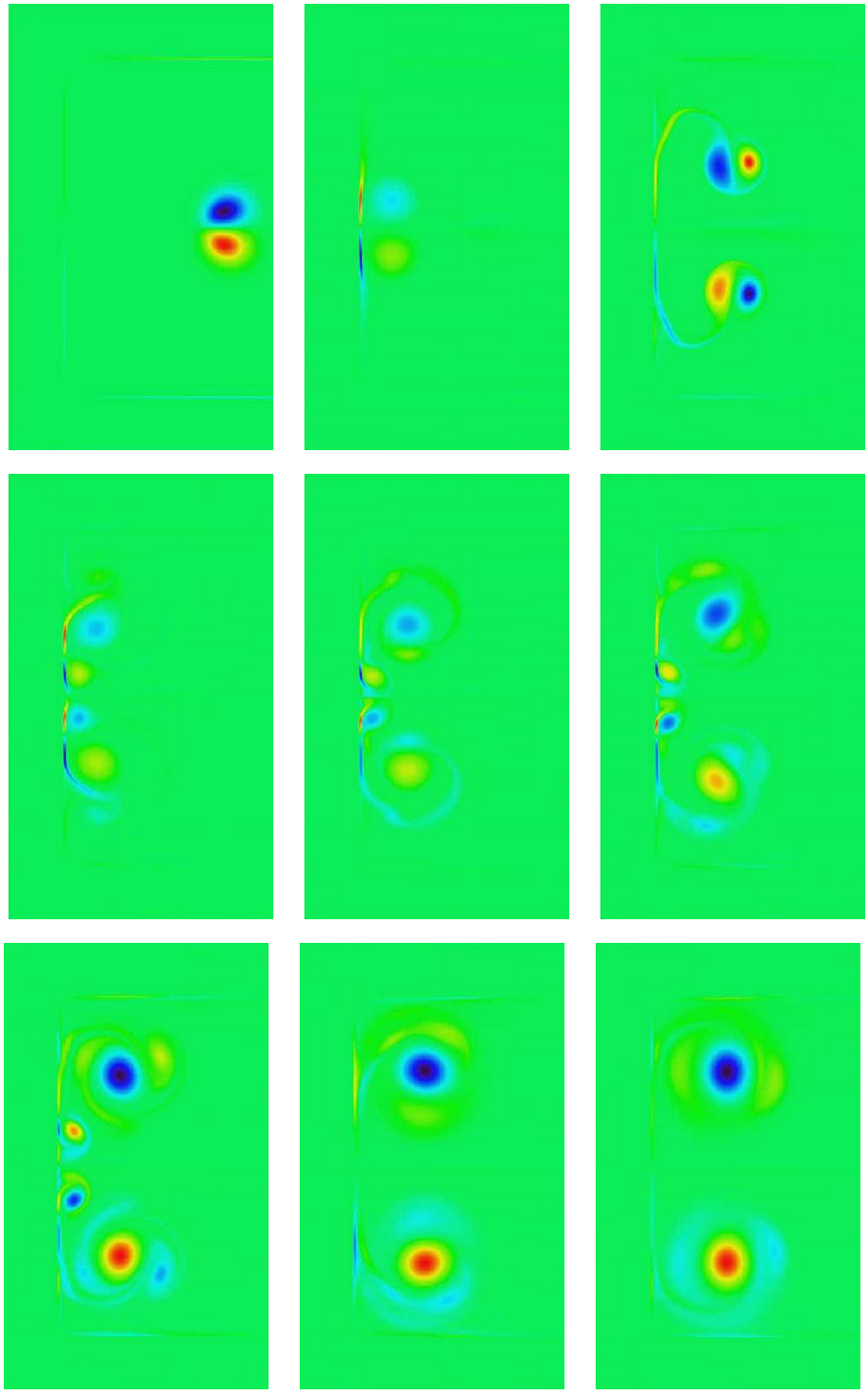


FIG. 3.5: Collision of a vortex dipole with a wall. A thin boundary layer develops and new vortices are formed.

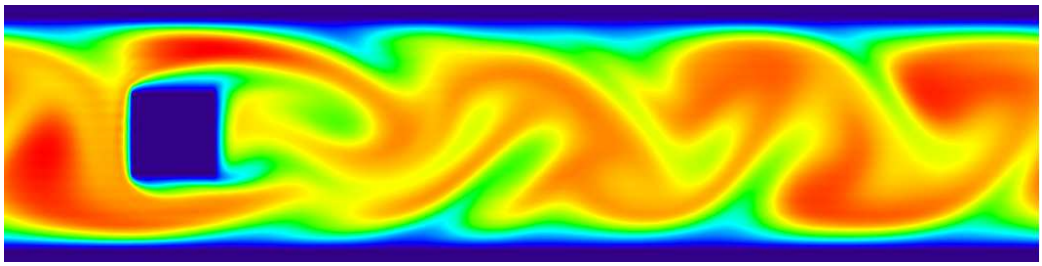


FIG. 3.6: Advection of a passive scalar in a von Kàrmàn vortex street behind a box.

atmosphere are among the most prominent ones. *Passive* refers to the fact, that the advected quantity has no feedback on the carrier flow and *scalar* just indicates that a scalar quantity is considered. From these facts it is easy to guess, that the scalar field $T(\boldsymbol{x}, t)$ evolves according to a advection-diffusion equation

$$\frac{\partial}{\partial t} T + \boldsymbol{u} \cdot \nabla T = \kappa \Delta T, \tag{3.19}$$

where κ denotes the diffusivity of the scalar quantity. $\boldsymbol{u}$ is the velocity field advecting the scalar, which has to be calculated from the vorticity equation. Formally this equation is similar to the two-dimensional vorticity equation, however the important difference is, that $\boldsymbol{u}$ is independent from T, hence a differing dynamics has to be expected.

The numerical implementation is straightforward. In addition to the vorticity equation a scalar field has to be simulated. The nonlinearity is calculated analogously to the one of the vorticity equation, and the velocity field in real space may be recycled. Boundary conditions can be implemented with the above presented penalization method, such that the equations of motion in Fourier space finally read

$$\frac{\partial}{\partial t} \tilde{\omega} = i\boldsymbol{k} \times \mathcal{F}\big[\boldsymbol{u} \times \boldsymbol{\omega} - \frac{1}{\epsilon} H \boldsymbol{u}\big] - \nu k^2 \tilde{\omega} \tag{3.20}$$

$$\frac{\partial}{\partial t} \tilde{T} = i\boldsymbol{k} \times \mathcal{F}\big[\boldsymbol{u} \times \boldsymbol{T} - \frac{1}{\epsilon} H \boldsymbol{u}\big] - \mathcal{F}\big[\frac{1}{\epsilon} H \boldsymbol{u}\big] - \kappa k^2 \tilde{T}, \tag{3.21}$$

where $\boldsymbol{T} = T\boldsymbol{e}_z$. Due to the additional diffusivity κ an additional CFL-constraint has to be included when explicit time stepping schemes are considered.

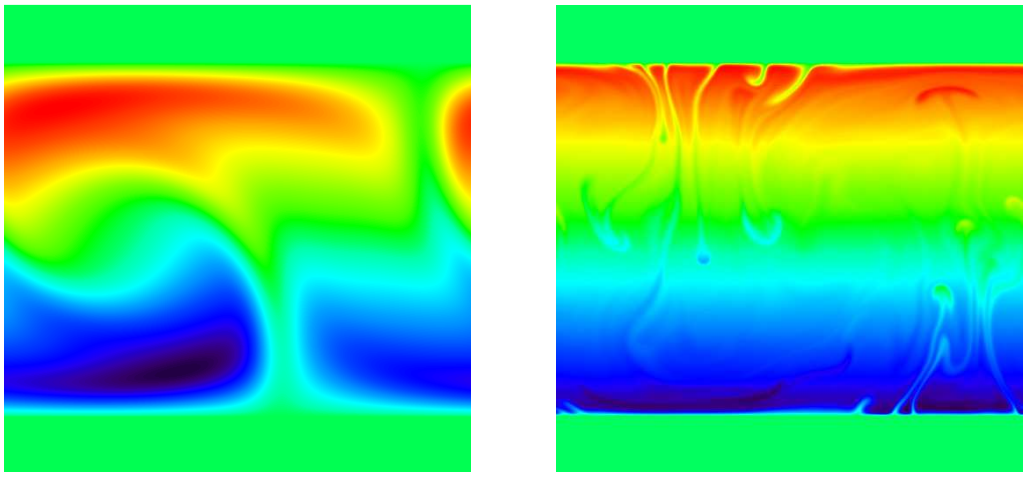


FIG. 3.7: Left picture: Convection rolls in the Θ profile of the Rayleigh-Bènard system. Right picture: High Rayleigh Number plumulence.

3.4 The Rayleigh-Bènard Problem

The Rayleigh-Bènard Problem is a well-studied hydrodynamic problem often encountered in natural and technical contexts. It serves as a paradigm for a complex system. It involves an abundant number of degrees of freedom represented by the flow field, which interact in a collective manner yielding a rich spectrum of possible solutions ranging from quiescent ground states over ordered convection cells to developed turbulence. The basic setup is very easy. A fluid is enclosed between two horizontal plates. The upper plate is cooled whereas the lower one is heated. Additionally the fluid is subject to gravitational forces. Now the change of temperature induces density fluctuations, which result in buoyancy forces. This produces an inherently unstable situtation: the lower fluid regions are heated an hence less dense than the upper, cool regions. The system is stabilized by viscous forces, which suppress an onset of convection. Below a critical temperature gradient the fluid is at rest, above this threshold convection sets in. We will not go into deeper details here and refer the reader to the broad literature on the subject, which provides further information to the physical problem (see [13, 11] for an introduction).

3.4.1 The Basic Equations in Boussinesq-Oberbeck Approximation

The basic equations are presented within the Boussinesq-Oberbeck approximation, which yields accurate results in many typical situations. The idea of this approximation is that the influence of density fluctuations is only taken into account for the volume specific forces and neglected for all other terms. For this approximation the Navier-Stokes equations take the form

$$\frac{\partial}{\partial t}\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p - \frac{\rho}{\rho_0} g\boldsymbol{e}_z + \nu\Delta\boldsymbol{u}. \tag{3.22}$$

The pressure is determined by the incompressibility condition $\nabla\cdot\boldsymbol{u} = \boldsymbol{0}$. Next comes the equation of state for the density field,

$$\delta\rho = \rho(T) - \rho(T_0) = -\alpha(T - T_0). \tag{3.23}$$

Here T_0 is the temperature of the lower plate and α denotes the thermal expansion coefficient of the fluid. The density depends on the temperature field, which obeys a advection-diffusion equation according to

$$\frac{\partial}{\partial t}T + \boldsymbol{u}\cdot\nabla T = \kappa\Delta T. \tag{3.24}$$

That means, the flow is governed by the highly non-linear Navier-Stokes equations and include gravitational forces depending on the fluid density. The density then depends on the local temperature field, which is coupled to the velocity field of the flow. Together with the boundary conditions (for which we will choose no-slip b.c. in the following) and proper initial conditions the system is fully specified.

3.4.2 New Variables

As can be verified with a short calculation, the motionless ground state is described by [11]

$$\boldsymbol{u}^{(s)} = \boldsymbol{0} \tag{3.25}$$

$$T^{(s)} = T_0 - \beta z \tag{3.26}$$

$$p^{(s)} = \rho_0 g z(1 + \alpha\beta z) \tag{3.27}$$

where β is related to the temperature gradient. Now new variables are introduced describing deviations from this ground state according to [11]

$$\boldsymbol{u} = \boldsymbol{u}^{(s)} + \boldsymbol{u}' \tag{3.28}$$

$$T = T^{(s)} + \Theta \tag{3.29}$$

$$p = p^{(s)} + \rho_0 \tilde{p} \tag{3.30}$$

Insertion into the basic equations and dropping the primes leads to

$$\frac{\partial}{\partial t}\boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla \tilde{p} - \alpha g \Theta \boldsymbol{e}_z + \nu \Delta \boldsymbol{u} \tag{3.31}$$

$$\frac{\partial}{\partial t}\Theta + \boldsymbol{u} \cdot \nabla \Theta = \beta u_z + \kappa \Delta \Theta. \tag{3.32}$$

The boundary conditions for Θ now read $\Theta(0) = \Theta(d) = 0$, where d denotes the height of the system. The physical control parameters of the system are the Raleigh number Ra and the Prandtl number Pr, which are calculated via

$$Ra = \frac{\alpha g \beta d^4}{\kappa \nu} \tag{3.33}$$

$$Pr = \frac{\nu}{\kappa}. \tag{3.34}$$

3.4.3 Numerical Evaluation in Two Dimensions

The Rayleigh-Bènard system often is simulated with the help of finite volume or Chebychev methods. The introduction of new variables, however, allows to impose the necessary boundary conditions with the volume penalization technique and hence makes it accessible to pseudospectral methods. This may eventually proof to be useful in case of high Rayleigh number flows, where fine flow structures emerge within the whole simulation domain. The equidistant grid here might be an advantage.

For the simulation of the two-dimensional system the vorticity formulation is advantageous, as we only have to deal with a scalar quantity. The basic equations of motion together with imposed boundary conditions then take the form

$$\frac{\partial}{\partial t}\omega = \nabla \times [\boldsymbol{u} \times \boldsymbol{\omega} - \frac{1}{\epsilon} H \boldsymbol{u}] - \alpha g \frac{\partial}{\partial x}\Theta + \nu \Delta \omega \tag{3.35}$$

$$\frac{\partial}{\partial t}\Theta = \nabla \times [\boldsymbol{u} \times \boldsymbol{\omega}] - \frac{1}{\epsilon} H \Theta + \beta u_z + \kappa \Delta \Theta. \tag{3.36}$$

In Fourier space the equations are accordingly given by

$$\frac{\partial}{\partial t}\tilde{\omega} = i\boldsymbol{k} \times \mathcal{F}[\boldsymbol{u} \times \boldsymbol{\omega} - \frac{1}{\epsilon} H\boldsymbol{u}] - i\alpha g k_x \tilde{\Theta} - \nu k^2 \tilde{\omega} \tag{3.37}$$

$$\frac{\partial}{\partial t}\tilde{\Theta} = i\boldsymbol{k} \times \mathcal{F}[\boldsymbol{u} \times \boldsymbol{\omega}] - \mathcal{F}[\frac{1}{\epsilon} H\boldsymbol{u}] + \beta u_z - \kappa k^2 \tilde{\Theta}. \tag{3.38}$$

The velocity field is calculated by inversion of the Poisson equation in Fourier space,

$$\tilde{\boldsymbol{u}} = i\frac{\boldsymbol{k} \times \boldsymbol{\omega}}{k^2}. \tag{3.39}$$

Care has to be taken when using an explicit time stepping scheme, as the numerical stability now is subject to four constraints: $u_{\max}, \nu, \kappa$ and ϵ. Once embedded into a time stepping scheme the right-hand side is evaluated according to

- calculation of $\tilde{\boldsymbol{u}}$ from ω
- FFT to real space with $\boldsymbol{u}$ and copies of ω and Θ
- calculation of the nonlinear terms $\boldsymbol{u} \times \boldsymbol{\omega} - \frac{1}{\epsilon} H\boldsymbol{u}$, $\boldsymbol{u} \times \boldsymbol{\Theta}$ and $-\frac{1}{\epsilon} H\boldsymbol{\Theta}$
- FFT of the nonlinear terms back to Fourier space
- dealiasing of the nonlinear terms
- building of the complete right-hand side including nonlinear terms, diffusive terms and couplings.

.1 The fourth-order Runge–Kutta method

Let us consider an initial value problem for some function $x(t)$ of the form

$$\dot{x}(t) = f(t,\, x(t)), \qquad x(t_0) = x_0. \tag{40}$$

The function $f(t, x)$ here is continuously differentiable on some interval $[a,\, b]$. Our goal is to solve Eq. (40) on the time interval $t \in [0,\, T]$. To this aim let us divide the interval into

$$0 = t_0 < t_1 < t_2 < \ldots < t_N = T, \qquad t_{n+1} = t_n + h, \qquad n = 0,\, \ldots,\, N-1.$$

The family of explicit Runge–Kutta (RK) methods of the m'th order is given by

$$x(t_{n+1}) := x_{n+1} = x_n + h \sum_{i=1}^{m} c_i k_i, \tag{41}$$

where

$$\begin{aligned}
k_1(t,\, x) &= f(t,\, x), \\
k_2(t,\, x) &= f(t + \alpha_2 h,\, x + h\beta_{21} k_1(t,\, x)), \\
k_3(t,\, x) &= f(t + \alpha_3 h,\, x + h(\beta_{31} k_1(t,\, x) + \beta_{32} k_2(t,\, x))), \\
&\vdots \\
k_m(t,\, x) &= f(t + \alpha_3 h,\, x + h \sum_{j=1}^{m-1} \beta_{mj} k_j(t,\, x)).
\end{aligned}$$

To specify a particular method, we need to specify the integer m (the number of stages), and the coefficients α_i (for $i = 2, 3, ..., m$), β_{ij} (for $1 \le j < i \le m$), and c_i (for $i = 1, 2, ..., m$). These data are usually arranged in a co-called *Butcher tableau* (after John C. Butcher):

$$\begin{array}{c|ccccc}
0 & & & & & \\
\alpha_2 & \beta_{21} & & & & \\
\alpha_3 & \beta_{31} & \beta_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
\vdots & \vdots & \vdots & & & \\
\alpha_m & \beta_{m1} & \beta_{m2} & \ldots\ldots & \beta_{mm-1} & \\
\hline
 & c_1 & c_2 & \ldots\ldots & c_{m-1} & c_m
\end{array}$$

One member of the family of Runge–Kutta methods is often referred to as *RK4 method* and corresponds to the one of the solutions corresponding to

the case $m = 4$. The corresponding Butcher tableau reads

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	1/3	1/3	1/6

The tableau above yields the equivalent corresponding equations defining the method:

$$x_{n+1} = x_n + \frac{h}{6}\big(k_1 + 2k_2 + 2k_3 + k_4\big), \tag{42}$$

where

$$\begin{aligned} k_1 &= f(t_n,\, x_n), \\ k_2 &= f(t_n + \frac{h}{2},\, x_n + \frac{h}{2}k_1), \\ k_3 &= f(t_n + \frac{h}{2},\, x_n + \frac{h}{2}k_2), \\ k_4 &= f(t_n + h,\, x_n + hk_3). \end{aligned}$$

This method is reasonably simple and robust and is a good general candidate for numerical solution of differential equations when combined with an intelligent adaptive step-size routine or an embedded methods (,e.g., Runge-Kutta-Fehlberg methods (RKF45)).

.2 The Runge–Kutta–Fehlberg methods

One way to guarantee accuracy in the solution of (40) is to solve the problem twice using step sizes h and $h/2$ and compare answers at the mesh points corresponding to the larger step size. But this requires a significant amount of computation for the smaller step size and must be repeated if it is determined that the agreement is not good enough. The Runge-Kutta-Fehlberg methods (RKF45) are one way to try to resolve this problem. At each step, two different approximations for the solution are made and compared. Usually an 4'th order method is used together with an 5'th order method that uses all of the points of the first one, and hence is often referred

to as an RKF45 method. The general form of a fifth-order Runge-Kutta is

$$\begin{aligned} k_1(t,\, x) &= f(t,\, x), \\ k_2(t,\, x) &= f(t + \alpha_2 h,\, x + h\beta_{21} k_1(t,\, x)), \\ &\vdots \\ k_6(t,\, x) &= f(t + \alpha_6 h,\, x + h \sum_{j=1}^{5} \beta_{6j} k_j(t,\, x)) \\ x_{n+1} &= x_n + h \sum_{i=1}^{6} c_i k_i + \mathcal{O}(h^6), \end{aligned}$$

The embedded fourth-order formula is

$$x^*_{n+1} = x_n + h \sum_{i=1}^{6} c^*_i k_i + \mathcal{O}(h^5),$$

The particular values for the cases of Runge-Kutta-Fehlberg-Parameters and Cash-Karp-Parameters are given below:

Runge-Kutta-Fehlberg-Parameters:

1/4	1/4					
3/8	3/32	9/32				
12/13	1932/2197	-7200/2197	7296/2197			
1	439/216	-8	3680/513	-845/4104		
1/2	-8/27	2	-3544/2565	1859/4104	-11/40	
	25/216	0	1408/2565	2197/4104	-1/5	
	16/135	0	6656/12825	28561/56430	-9/50	2/55

Cash-Karp-Parameters:

1/5	1/5					
3/10	3/40	9/40				
3/5	3/10	-9/10	6/5			
1	-11/54	5/2	-70/27	35/27		
7/8	1631/55296	175/512	575/13828	44275/110592	253/4096	
	37/378	0	250/621	125/594	512/1771	
	2825/27648	0	18575/48384	13525/55296	277/14336	1/4

The error estimate is

$$\Delta = x_{n+1} - x^*_{n+1} = \sum_{i=1}^{6}(c_i - c^*_i)k_i.$$

If we take the current step h and produce an error Δ, the corresponding "optimal" step h_0 is estimated as

$$h_0 = h\left|\frac{\Delta_{tol}}{\Delta}\right|^{0.2},$$

where Δ_{tol} is a disired accuracy. Then if the two answers are in close agreement, the approximation is accepted. If $\Delta > \Delta_{tol}$ the step size has to be decreased, whereas the relation $\Delta < \Delta_{tol}$ means, that the step size are to be increased in the next step.

In the practice the optimal step can be often written as [17]

$$h_0 = \begin{cases} sh\left|\frac{\Delta_{tol}}{\Delta}\right|^{0.20}, & \Delta \geq \Delta_{tol}, \\ sh\left|\frac{\Delta_{tol}}{\Delta}\right|^{0.25}, & \Delta < \Delta_{tol}, \end{cases}$$

where s is a "safety" factor, $s \simeq 0.8$.

.3 Ordering of Fourier Coefficients (Fortran)

Due to algorithmic details of the fast Fourier transform the Fourier coefficients are ordered in a somewhat "strange" manner, at least for a newbie. This causes problems especially when calculating derivatives of functions in Fourier space, as calculating the derivative corresponds to a multiplication with a real wave number (or vector in more than one spatial dimension). Hence the wave vectors have to be stored in a real array with exactly the same ordering as the Fourier coefficients. A complex field sampled by N grid points is transformed to a complex array of Fourier coefficients being ordered according to

$$\tilde{f}_0, \tilde{f}_1, \ldots, \tilde{f}_{\frac{N}{2}}, \tilde{f}_{-\frac{N}{2}+1}, \tilde{f}_{-\frac{N}{2}+2}, \ldots, \tilde{f}_{-1} \tag{43}$$

If we consider a real field f the Fourier coefficients exhibit a symmetry $\tilde{f}_{-j} = \tilde{f}_j^*$. This is exploited by most Fourier transform by only storing roughly half of the array, reducing the memory needed for holding the coefficients and giving a speed increase of roughly a factor of two. The Fourier coefficients in Fourier space are then ordered according to

$$\tilde{f}_0, \tilde{f}_1, \ldots, \tilde{f}_{\frac{N}{2}} \tag{44}$$

Let L denote the physical length of the simulation domain. The real array of wave numbers then takes the form

$$k(i) = \begin{cases} \frac{2\pi}{L} i & \text{if} \quad i = 0, \ldots, \frac{N}{2} \\ \frac{2\pi}{L}(-N+i) & \text{if} \quad i = \frac{N}{2}+1, \ldots, N-1 \end{cases} \tag{45}$$

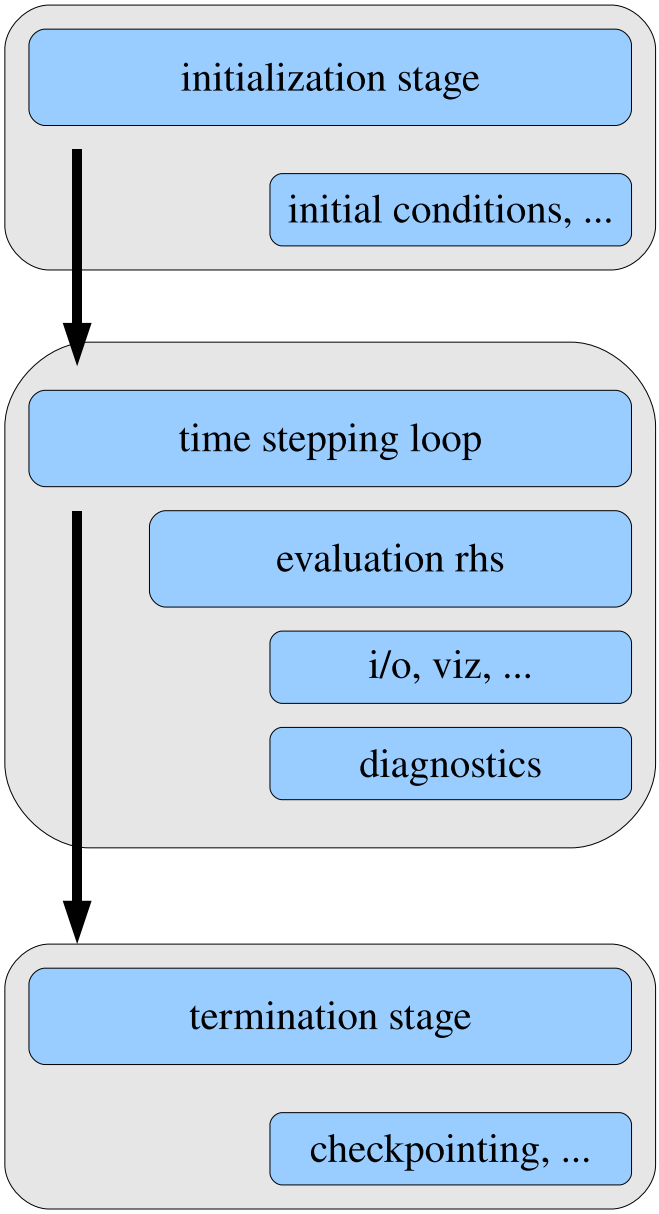
In case of a real field, the above-mentioned symmetry may be exploited. Then the array holding the wave vectors has also roughly half the size resulting in the index i running only $i = 0, \ldots, \frac{N}{2}$. These considerations are generalized to a two-dimensional field $f(\boldsymbol{x}, t)$ in a straight-forward manner. In this case we have a two-dimensional array for the Fourier coefficients. In case f represents a complex field, the wave vectors are arranged according to

$$k_x(i,j) = \begin{cases} \frac{2\pi}{L} i & \text{if} \quad i = 0, \ldots, \frac{N}{2} \\ \frac{2\pi}{L}(-N+i) & \text{if} \quad i = \frac{N}{2}+1, \ldots, N-1 \end{cases} \tag{46}$$

$$k_y(i,j) = \begin{cases} \frac{2\pi}{L} j & \text{if} \quad j = 0, \ldots, \frac{N}{2} \\ \frac{2\pi}{L}(-N+j) & \text{if} \quad j = \frac{N}{2}+1, \ldots, N-1 \end{cases}$$

In case of a real field, half of the Fourier coefficients suffice and the index i for the wave vectors reduces to $i = 0, \ldots, \frac{N}{2}$. One should note that different implementations of the fast Fourier algorithms exhibit a different ordering of Fourier coefficients. The ordering presented here is for example used by the `FFTW` library.

.4 Flow Diagram for a Typical Simulation Code



.5 Exponential Time Differencing und Integrating Factor Methods

We are interested in the solution of nonlinear evolution equation of the form

$$u_t = \mathcal{L}u + \mathcal{N}(u), \tag{47}$$

$u = u(x,t)$, $x \in \Omega \subset R^n$, $n = 1, 2, 3$ and $t \in [0,\, T]$. We also supply the initial conditions

$$u(x,0) = u_0(x) \quad \text{in}\,\Omega$$

and boundary conditions, i.e., periodic. The operators $\mathcal{L}$ and $\mathcal{N}$ denote linear and nonlinear parts, respectively.
Here we concentrate on time-discretization schemes with *exact* treatment of the *linear* part for solving nonlinear equation written above. That is, if the nonlinear term of the equation is zero, then the scheme reduces to the evalution of the exponential function of the operator representing the linear term.

.5.1 Exponential Time Differencing Methods (ETD)

In order to simplify the notation we replace the linear operator $\mathcal{L}$ by a scalar q, i.e.,

$$u_t = q\,u + \mathcal{N}(u,\,t). \tag{48}$$

First we multiply (48) by *the integrating factor* e^{-qt} and integrate the equation over a single time step from $t = t_n$ to $t = t_n + h$ and obtain the *exact* relation

$$\big(u(t_{n+1})e^{-qh} - u(t_n)\big)e^{-qt_n} = \int_{t_n}^{t_n+h} e^{-qt}\mathcal{N}(u,\,t)dt \tag{49}$$

or, equivalently,

$$u(t_{n+1}) = u(t_n)e^{qh} + e^{qh}\int_0^h e^{-q\tau}\mathcal{N}(u(x,t_n+\tau),t_n+\tau)d\tau. \tag{50}$$

The difference between different ETD methods consist in difference approximations to the integral in the equation above.

The simplest approximation to the integral in (50) is that $\mathcal{N}$ is constant between $t = t_n$ and $t = t_{n+1} := t_n + h$, i.e.,

$$\mathcal{N} = \mathcal{N}_n + \mathcal{O}(h),$$

where $u_n = u(t_n)$ and $\mathcal{N}_n = \mathcal{N}(u_n, t_n)$. Then Eq. (50) becomes the scheme **ETD1**, given by

$$u_{n+1} = u_n e^{qh} + \mathcal{N}_n \frac{e^{qh} - 1}{q}. \tag{51}$$

Now let us consider the higher-order approximation of the form

$$\mathcal{N} = \mathcal{N}_n + \tau \frac{\mathcal{N}_n - \mathcal{N}_{n-1}}{h} + \mathcal{O}(h^2).$$

Then one obtains the so-called scheme **ETD2**

$$u_{n+1} = u_n e^{qh} + \mathcal{N}_n \frac{(1+hq)e^{qh} - 1 - 2hq}{hq^2} + \mathcal{N}_{n-1} \frac{-e^{qh} + 1 + hq}{hq^2}. \tag{52}$$

Notice that ETD schemes of arbitary order can also be derived [9]. Other possibility is to use ETD schemes, combined with Runge-Kutta methods [9] or so-called Integrating Factor Methods, briefly discussed below.

.5.2 Integrating Factor Methods (IFM)

The method of Integrating Factors (IFM) is also based on the idea that the problem in question can be transformed so that the linear part of the system is solved exactly. Integrating factor methods are usually obtained by rewriting (48) as [6, 9]

$$\frac{d}{dt}\left(ue^{-qt}\right) = e^{-qt}\mathcal{N}(u). \tag{53}$$

and then applying a time-stepping scheme to this equation.

For example, the forward Euler approximation reduces to

$$u_{n+1} = e^{qh}\Big(u_n + h\mathcal{N}(u_n)\Big) \tag{54}$$

In the same manner IFM can be embedded into different Runge-Kutta schemata. We mention only RK2 (Heun-Method):

$$u_{n+1} = u_n e^{qh} + \frac{h}{2}\Big(\mathcal{N}_n e^{qh} + \mathcal{N}((u_n + h\mathcal{N}_n)e^{qh},\, t+h)\Big) \tag{55}$$

Other IFM schemata are discussed in, e.g., [6, 9] in more details.

Bibliography

[1] I. S. Aranson and L. Kramer. The world of the complex ginzburg-landau equation. *Reviews of Modern Physics*, 74:99–143, 2002.

[2] W. van Saarloos P. C. Hohenberg H. Chaté B. I. Shraiman, A. Pumir and M. Holen. Spatiotemporal chaos in the one-dimensional complex ginzburg-landau equation. *Physica D*, 57:241–248, 1992.

[3] N. Bekki and K. Nozaki. Formation of spatial patterns and holes in the generalized complex ginzburg-landau equation. *Physics Letters*, 110A(3):133–135, 1985.

[4] Michael Bestehorn. *Hydrodynamik und Strukturbildung. Mit einer kurzen Einführung in die Kontinuumsmechanik*. Springer, Berlin, 1 edition, September 2006.

[5] John P. Boyd. *Chebyshev and Fourier Spectral Methods: Second Revised Edition*. Dover Publications, 2 revised edition, December 2001.

[6] Claudio Canuto, M.Yousuff Hussaini, Alfio Quarteroni, and Thomas A. Jr. Zang. *Spectral Methods in Fluid Dynamics*. Springer, July 1993.

[7] H. Chaté. Spatiotemporal intermittency regimes of the one-dimensional complex ginzburg-landau equation. *Nonlinearity*, 7:185–204, 1994.

[8] H. Chaté and P. Manneville. Phase diagram of the two-dimensional complex ginzburg-landau equation. *Physica A*, 224:348–368, 1996.

[9] S. M. Cox and P. C. Matthews. Exponential time differencing for stiff systems. *Journal of Computational Physics*, 176:430–455, 2002.

[10] G.B. Ermentrout. Stable small-amplitude solutions in reaction-diffusion systems. *Quarterly of Applied Mathematics*, 39:61–86, 1981.

[11] H. Haken. *Synergetics – Introduction and Advanced Topics*. Springer-Verlag, Berlin, 1983.

[12] T. Y. Hou and R. Li. Computing nearly singular solutions using pseudo-spectral methods. *J. Comp. Phys*, 226:379 – 397, 2007.

[13] M. Haase J. Argyris, G. Faust. *Die Erforschung des Chaos*. Vieweg Verlag, Braunschweig, 1995.

[14] G. H. Keetels, U. D'Ortona, W. Kramer, H. J. H. Clercx, K. Schneider, and G. J. F. van Heijst. Fourier spectral and wavelet solvers for the incompressible navier-stokes equations with volume-penalization: Convergence of a dipole-wall collision. *J. Comput. Phys.*, 227(2):919–945, 2007.

[15] J. Lega. Travelling hole solutions of the complex ginzburg-landau equation: a review. *Physica D*, 152-153:269–287, 2001.

[16] A.C. Newell and J.A. Whitehead. Finite bandwidth, finite amplitude convection. *The Journal of Fluid Mechanics*, 38:279–303, 1969.

[17] William H. Press, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, www.nr.com, 1993.

[18] Kai Schneider. Numerical simulation of the transient flow behaviour in chemical reactors using a penalisation method. *Computers & Fluids*, 34(10):1223 – 1238, 2005.

[19] L.A. Segel. Distant side-walls cause slow amplitude modulation of cellular convection. *The Journal of Fluid Mechanics*, 38:203–224, 1969.

[20] C. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comp. Phys.*, 77(12):439 – 397, 1988.

[21] J.T. Stuart. On the non-linear mechanics of wave disturbances in stable and unstable parallel flows. part 1. the basic behaviour in plane poiseuille flow. *The Journal of Fluid Mechanics*, 9:353–370, 1960.

[22] Lloyd N. Trefethen. *Spectral Methods in MATLAB*. SIAM: Society for Industrial and Applied Mathematics, illustrated edition edition, February 2001.

[23] W. van Saarloos and P. C. Hohenberg. Fronts, pulses, sources and sinks in generalized complex ginzburg-landau equation. *Physica D*, 56:303–367, 1992.