2

Numerical implementation

2.1 Numerical approach: choice of the method

The choice of the numerical method is always a difficult task. In our case the main part of the work is heavily supported by the numerical approximation of system (1.7) and boundary conditions (1.6): this can give an idea of their importance. For the spatial discretization of the channel we have adopted spectral methods and for the temporal discretization we use conventional finite differences.

Spectral methods make use of global representation of functions, usually by high order polynomials or Fourier series, in contrast with finite differences or finite elements in which the representation is local. With a properly designed spectral method, if the approximated solution is infinitely differentiable, errors go to zero faster than any negative power of the number of retained modes (cf. proposition B.1). Instead, finite differences and finite elements only yield finite order rates of convergence and thus the spatial resolution must be increased so as to get comparable precision to spectral methods. Besides, spectral methods also have great resolution in boundary layers as the ones close to the channel walls in our situation.

One of the main source of difficulties of spectral methods are irregular domains, but that is not the case for the channel flow. The possibility of using fast Fourier transform have made spectral methods suitable for fluid problems where high accuracy is important to simulate complicated solutions. Typical discretizations employ Fourier series for periodic boundary conditions, as in the stream direction in our model, and Chebyshev polynomials for rigid boundary conditions, on the channel walls in our case. For both approximations the possibility of applying fast Fourier transforms suppose a great improvement in the computations.

Let us now describe the numerical procedure. We want to follow the temporal evolution of an initial flow subjected to the incompressibility condition, $\boldsymbol{\nabla} \cdot \mathbf{u} = 0$, and boundary conditions (1.6). To this end we use a spectral method to approximate velocities u, v and pressure deviation p', which from now on we consider non-dimensional quantities. We recall from (1.3) that $p = p' - Gx$

and from (1.9) and table 1.2.

$$G = -\frac{1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1} \quad \text{or} \quad G = \frac{2}{Re_p},$$

respectively for the constant flux or pressure cases, so in the first case the mean pressure gradient varies with time and it is constant for the second one. The approximation chosen of the periodic variable x relies on Galerkin method and Fourier series (see §B.1).

Galerkin method. Since we are supposing u, v, p' periodic in x, we replace them by their Fourier series truncated at some positive integer N:

$$(u, v, p')(x, y, t) = \sum_{k=-N}^{N} (\hat{u}_k, \hat{v}_k, \hat{p}_k)(y, t)\mathrm{e}^{\mathrm{i}k\alpha x}, \tag{2.1}$$

for $x \in \mathbb{R}$, $y \in [-1, 1]$ and $t \geqslant 0$. This finite series substituted in system (1.7) eliminates x-derivatives and gives rise to a system of partial differential equations in variables y and t for the Fourier coefficients $\hat{u}_k, \hat{v}_k, \hat{p}_k$,

$$\begin{cases} \dfrac{\partial \hat{u}_k}{\partial t} + \left[(u-c)\widehat{\dfrac{\partial u}{\partial x}} + v\dfrac{\partial u}{\partial y}\right]_k = -\mathrm{i}k\alpha\hat{p}_k + \dfrac{1}{Re}\left(-k^2\alpha^2\hat{u}_k + \dfrac{\partial^2 \hat{u}_k}{\partial y^2}\right) \\ \qquad\qquad + \delta_{k0}G, \\ \dfrac{\partial \hat{v}_k}{\partial t} + \left[(u-c)\widehat{\dfrac{\partial v}{\partial x}} + v\dfrac{\partial v}{\partial y}\right]_k = -\dfrac{\partial \hat{p}_k}{\partial y} + \dfrac{1}{Re}\left(-k^2\alpha^2\hat{v}_k + \dfrac{\partial^2 \hat{v}_k}{\partial y^2}\right), \\ 0 = \mathrm{i}k\alpha\hat{u}_k + \dfrac{\partial \hat{v}_k}{\partial y}, \end{cases} \tag{2.2}$$

where $-N \leqslant k \leqslant N$, $\widehat{[\cdot]}_k$ stands for the kth-order Fourier coefficient of $[\cdot]$, $\delta_{00} = 1$, and $\delta_{k0} = 0$ for $k \neq 0$. This constitutes the spectral Fourier–Galerkin method, a process which may be also viewed as the orthogonal projection by means of the scalar product of $L^2[0, L]$, over the space generated by $\exp(\mathrm{i}k\alpha x)$ for $k = -N, \ldots, N$. The projected object consists of the residual obtained upon the substitution of the previously defined truncated series in system (1.7).

As u is supposed to be a real function, it follows that $\hat{u}_k^* = \hat{u}_{-k}$ ('*' denotes complex conjugate) and analogously for v and p'. Therefore we may write (2.2) for negative modes $k = -N, \ldots, -1$ in terms of nonnegative modes $k = 0, \ldots, N$ and their conjugates. It is enough then to consider modes in $\hat{u}_k, \hat{v}_k, \hat{p}_k$ and equations in (2.2) for $k = 0, \ldots, N$. For the no-slip boundary conditions in (1.6) it turns out

$$0 = u(x, \pm 1, t) = \sum_{k=-N}^{N} \hat{u}_k(\pm 1, t)\mathrm{e}^{\mathrm{i}k\alpha x}, \quad \text{for every } x \in \mathbb{R},$$

and applying the same to v, from proposition B.2 in appendix B, we obtain

$$(\hat{u}_k, \hat{v}_k)(\pm 1, t) = 0, \quad t \geqslant 0, \quad k = 0, \ldots, N. \tag{2.3}$$

Collocation method. To get rid of derivatives in the transversal variable y, we employ the collocation method, in which every equation is imposed at selected (collocation) points: we have chosen the Chebyshev abscissas because of their good convergence properties and the possibility of utilizing fast Fourier transforms as previously mentioned (see §B.2 for more details).

We express the Fourier coefficients $\hat{u}_k, \hat{v}_k, \hat{p}_k$ for $k = 0, \ldots, N$, as a truncated Chebyshev series

$$\hat{u}_k(y,t) = \sum_{j=0}^{M} \tilde{u}_{kj}(t) T_j(y), \qquad \hat{v}_k(y,t) = \sum_{j=0}^{M} \tilde{v}_{kj}(t) T_j(y),$$
$$\hat{p}_k(y,t) = \sum_{j=0}^{M-1} \tilde{p}_{kj}(t) T_j(y),$$

being $T_j(y) = \cos(j \arccos(y))$ for $j = 0, \ldots, M$ the Chebyshev polynomials, and M a positive integer. In contrast to Galerkin method, in this case we interpolate these truncated series at selected collocation points. The choice of those points is detailed in §B.2. In our case we need two different sets of collocation points

$$\begin{aligned} y_m &= \cos(\pi m/M), && m = 0, \ldots, M, \\ \bar{y}_m &= \cos(\pi(m+1/2)/M), && m = 0, \ldots, M-1, \end{aligned} \tag{2.4}$$

and we define the discrete unknowns of the system as

$$\begin{aligned} \hat{u}_{km}(t) = \hat{u}_k(y_m, t), \qquad & \hat{v}_{km}(t) = \hat{v}_k(y_m, t), && m = 0, \ldots, M, \\ & \hat{p}_{km}(t) = \hat{p}_k(\bar{y}_m, t), && m = 0, \ldots, M-1. \end{aligned} \tag{2.5}$$

To obtain a system of ordinary differential equations in t, momentum equations in (2.2) are enforced on the first grid, y_m, whereas the second one, $\bar{y}_m$, is taken for the continuity equation. If we used the same collocation points for the pressure, $\hat{p}_k$ and continuity equation as for the velocity, $\hat{u}_k, \hat{v}_k$ and momentum equations, then we would obtain an undetermined linear system for the discrete dependent variables $\hat{u}_{km}, \hat{v}_{km}, \hat{p}_{km}$ (cf. Canuto, Hussaini, Quarteroni & Zang 1988, p. 295, for a theoretical point of view). The reason for this is that the gradient of the pressure mode $\tilde{p}_{0M}(t) T_M(y)$ vanishes at the points $y_m,\ m = 1, \ldots, M-1$ and thus $\tilde{p}_{0M}$ has no effect upon the velocity in the momentum equations. Indeed

$$\frac{\partial}{\partial x}(\tilde{p}_{0M}(t) T_M(y)) = 0, \qquad \frac{\partial}{\partial y}(\tilde{p}_{0M}(t) T_M(y)) = \tilde{p}_{0M}(t) T'_M(y),$$

but from $T_M(y) = \cos(M \arccos(y))$ we get

$$T'_M(y) = \frac{M \sin(M \arccos(y))}{\sqrt{1-y^2}} \quad \Longrightarrow \quad T'_M(y_m) = 0, \quad \text{for } m = 1, \ldots, M-1.$$

There is also another spurious mode, $\tilde{p}_{00}(t) T_0(y)$, which is related to the mean value of the pressure and it will be reviewed in §2.4.

Boundary conditions (2.3) are easily imposed, as for $k = 0, \dots, N$

$$\begin{aligned} (\hat{u}_k, \hat{v}_k)(\pm 1, t) = 0 \quad &\iff \quad (\hat{u}_k, \hat{v}_k)(y_0, t) = (\hat{u}_k, \hat{v}_k)(y_M, t) = 0 \\ &\iff \quad \hat{u}_{k0}(t) = \hat{u}_{kM}(t) = \hat{v}_{k0}(t) = \hat{v}_{kM}(t) = 0, \end{aligned} \tag{2.6}$$

and thus on the first grid y_m, we only consider $m = 1, \dots, M-1$ in (2.2) for momentum equations and unknowns $\hat{u}_{km}, \hat{v}_{km}$. The evaluation of (2.2) at the respective grids y_m and $\bar{y}_m$ gives rise to a system of differential-algebraic equations in t with $(2N+1)(3M-2)$ real equations and unknowns.

2.2 Evaluation of linear terms

We analyse the evaluation of linear terms in (2.2) by means of cosine transforms described below. To be precise we are referring to the following terms

$$\begin{aligned} &\frac{\partial^2 \hat{u}_k}{\partial y^2}(y_m), \quad \frac{\partial^2 \hat{v}_k}{\partial y^2}(y_m), \quad \hat{p}_k(y_m), \quad \frac{\partial \hat{p}_k}{\partial y}(y_m), && \text{on the first grid,} \\ &\qquad\qquad\qquad\quad \hat{u}_k(\bar{y}_m), \quad \frac{\partial \hat{v}_k}{\partial y}(\bar{y}_m), && \text{on the second grid.} \end{aligned}$$

We suppose that the values of the discrete unknowns defined in (2.5), are given. The outline of the process consists of the construction of the Chebyshev interpolating polynomial at the given values of the unknowns on its own grid, the computation of analytic derivatives for this polynomial if necessary, and finally the evaluation of the resulting polynomial at the appropriate grid. Let us detail those steps.

Interpolation of the first grid. Given values $w_0, \dots, w_M$ at the points $y_0, \dots, y_M$ on the first grid, then the interpolating Chebyshev polynomial $w(y)$ such that $w(y_m) = w_m$ for $m = 0, \dots, M$ is computed by

$$w(y) = \sum_{j=0}^{M} \tilde{w}_j T_j(y), \qquad \tilde{w}_j = \frac{2}{M \bar{c}_j} \sum_{m=0}^{M} \frac{w_m}{\bar{c}_m} \cos \frac{\pi j m}{M},$$

and $\bar{c}_0 = \bar{c}_M = 2$, $\bar{c}_j = 1$ if $j \neq 0, M$. These formulae are proved in theorem B.12. Conversely, from $\tilde{w}_0, \dots, \tilde{w}_M$ we recover $w_m = w(y_m)$,

$$w_m = \sum_{j=0}^{M} \tilde{w}_j T_j(y_m) = \sum_{j=0}^{M} \tilde{w}_j \cos \frac{\pi j m}{M}.$$

These are two linear transforms which can be abbreviated as

$$\begin{aligned} \tilde{w} &= C_1 w, && (C_1)_{jm} = \frac{2}{M \bar{c}_j \bar{c}_m} \cos \frac{\pi j m}{M}, \\ w &= C_1^{-1} \tilde{w}, && (C_1^{-1})_{jm} = \cos \frac{\pi j m}{M}, \end{aligned} \tag{2.7}$$

being $w = (w_0, \dots, w_M)^t$ and $\tilde{w} = (\tilde{w}_0, \dots, \tilde{w}_M)^t$. Actually we only need to consider $w = (w_1, \dots, w_{M-1})^t$ due to (2.6). Thus C_1 and C_1^{-1} can be considered as matrices of dimensions $(M+1) \times (M-1)$ and $(M-1) \times (M+1)$ respectively.

Interpolation of the second grid. Analogously if we want to interpolate $w_0, \dots, w_{M-1}$ at the points $\bar{y}_0, \dots, \bar{y}_{M-1}$ on the second grid, the interpolating Chebyshev polynomial $w(y)$ such that $w(\bar{y}_m) = w_m$ for $m = 0, \dots, M-1$ satisfies

$$w(y) = \sum_{j=0}^{M-1} \tilde{w}_j T_j(y), \qquad \tilde{w}_j = \frac{2}{M c_j} \sum_{m=0}^{M-1} w_m \cos \frac{\pi j(2m+1)}{2M},$$

and $c_0 = 2$, $c_j = 1$ if $j \neq 0$, as it is shown in theorem B.9. For the inverse transform we get

$$w_m = \sum_{j=0}^{M-1} \tilde{w}_j T_j(\bar{y}_m) = \sum_{j=0}^{M-1} \tilde{w}_j \cos \frac{\pi j(2m+1)}{2M}.$$

The abbreviation of the linear transforms is now

$$\begin{aligned} \tilde{w} &= C_2 w, & (C_2)_{jm} &= \frac{2}{M c_j} \cos \frac{\pi j(2m+1)}{2M}, \\ w &= C_2^{-1} \tilde{w}, & (C_2^{-1})_{jm} &= \cos \frac{\pi j(2m+1)}{2M}, \end{aligned} \tag{2.8}$$

being $w = (w_0, \dots, w_{M-1})^t$ and $\tilde{w} = (\tilde{w}_0, \dots, \tilde{w}_{M-1})^t$. The dimensions of matrices C_2 and C_2^{-1} is $M \times M$.

Derivative of Chebyshev polynomials. The last step in the calculation of linear terms involves evaluation of derivatives.

Proposition 2.1. *Let us suppose that $w(y)$, $w'(y)$ and $w''(y)$ can be expanded in Chebyshev series*

$$w(y) = \sum_{j=0}^{\infty} \tilde{w}_j T_j(y), \qquad w'(y) = \sum_{j=0}^{\infty} \tilde{w}'_j T_j(y), \qquad w''(y) = \sum_{j=0}^{\infty} \tilde{w}''_j T_j(y).$$

Then, for $j \geqslant 0$, the relations between coeffi cients are given by the recurrences

$$\begin{aligned} c_j \tilde{w}'_j &= \tilde{w}'_{j+2} + 2(j+1)\tilde{w}_{j+1}, \\ c_j \tilde{w}''_j &= \tilde{w}''_{j+2} + 2(j+1)\tilde{w}'_{j+1}, \end{aligned} \tag{2.9}$$

and also by the formulae

$$c_j \tilde{w}'_j = \sum_{\substack{m=j+1 \\ m+j \text{ odd}}}^{\infty} 2m\tilde{w}_m, \qquad c_j \tilde{w}''_j = \sum_{\substack{m=j+2 \\ m+j \text{ even}}}^{\infty} m(m^2 - j^2)\tilde{w}_m.$$

Proof: If we let $\theta = \arccos y$ then

$$T'_j(y) = \frac{j \sin(j \arccos(y))}{\sqrt{1-y^2}} = \frac{j \sin j\theta}{\sin \theta},$$

and therefore the trigonometric identity $2\sin\theta\cos j\theta = \sin(j+1)\theta - \sin(j-1)\theta$ can be translated to

$$2T_1(y) = \frac{T_2'(y)}{2}, \qquad 2T_j(y) = \frac{T_{j+1}'(y)}{j+1} - \frac{T_{j-1}'(y)}{j-1}, \quad j \geqslant 2. \tag{2.10}$$

Deriving formally the expansion of $w(y)$, equating to the expansion of $w'(y)$ and using (2.10) we obtain

$$\begin{aligned} w' &= \sum_{j=1}^{\infty} \tilde{w}_j T_j' = \sum_{j=0}^{\infty} \tilde{w}_j' T_j = \tilde{w}_0' T_1' + \tilde{w}_1' \frac{T_2'}{2} + \frac{1}{2} \sum_{j=2}^{\infty} \tilde{w}_j' \left(\frac{T_{j+1}'}{j+1} - \frac{T_{j-1}'}{j-1} \right) \\ &= \sum_{j=1}^{\infty} \frac{c_{j-1}\tilde{w}_{j-1}' - \tilde{w}_{j+1}'}{2j} T_j', \end{aligned}$$

so comparing term by term in the first and last series yields

$$2j\tilde{w}_j = c_{j-1}\tilde{w}_{j-1}' - \tilde{w}_{j+1}', \qquad j \geqslant 1.$$

Hence it follows (2.9) for w', which is easily extended to w''. We employ (2.9) for $j \geqslant 0$ to finally deduce

$$\begin{aligned} c_j\tilde{w}_j' &= \sum_{\substack{m=j+1 \\ m+j \text{ odd}}} \left(c_{m-1}\tilde{w}_{m-1}' - \tilde{w}_{m+1}' \right) = \sum_{\substack{m=j+1 \\ m+j \text{ odd}}} 2m\tilde{w}_m, \\ c_j\tilde{w}_j'' &= \sum_{\substack{m=j+1 \\ m+j \text{ odd}}} \left(c_{m-1}\tilde{w}_{m-1}'' - \tilde{w}_{m+1}'' \right) = \frac{1}{2} \sum_{\substack{m=j+1 \\ m+j \text{ odd}}} 4m\tilde{w}_m' \\ &= \frac{1}{2} \sum_{\substack{m=j+1 \\ m+j \text{ odd}}} \left[\left((m+1)^2 - j^2 \right) - \left((m-1)^2 - j^2 \right) \right] \tilde{w}_m' \\ &= \frac{1}{2} \sum_{\substack{m=j+2 \\ m+j \text{ even}}} (m^2 - j^2)(c_{m-1}'\tilde{w}_{m-1}' - \tilde{w}_{m+1}') = \sum_{\substack{m=j+2 \\ m+j \text{ even}}} m(m^2 - j^2)\tilde{w}_m. \end{aligned}$$

□

From the theorem, when $w(y)$ is represented as a finite series, its first and second derivatives can be represented by a matrix-vector product

$$\begin{aligned} \tilde{w}' &= D_y\tilde{w}, \quad (D_y)_{jm} = \begin{cases} 2m, & \text{if } m > j \text{ and } m+j \text{ odd} \\ 0, & \text{if } m \leqslant j \text{ or } m+j \text{ even} \end{cases} \\ \tilde{w}'' &= D_y^2\tilde{w}, \quad (D_y^2)_{jm} = \begin{cases} m(m^2 - j^2), & \text{if } m > j+1 \text{ and } m+j \text{ even} \\ 0, & \text{if } m \leqslant j+1 \text{ or } m+j \text{ odd} \end{cases} \end{aligned} \tag{2.11}$$

being $\tilde{w} = (\tilde{w}_0, \ldots, \tilde{w}_M)^t$, $\tilde{w}' = (\tilde{w}_0', \ldots, \tilde{w}_{M-1}', 0)^t$ and $\tilde{w}'' = (\tilde{w}_0'', \ldots, \tilde{w}_{M-2}'', 0, 0)^t$. Thus D_y and D_y^2 can be considered as matrices of dimension $(M+1) \times (M+1)$.

Gathering matricial operations (2.7), (2.8) and (2.11), and taking '$\widehat{\ }$' out of $\widehat{[\cdot]}_k$, $\hat{u}_k$, $\hat{v}_k$ and $\hat{p}_k$ for convenience, for $k = 0, \ldots, N$ we may write system (2.2) as

$$\dot{u}_k = -\left[(u-c)\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right]_k - D_{xk}C_1^{-1}C_2 p_k + \frac{1}{Re}(D_{xk}^2 + C_1^{-1}D_y^2 C_1)u_k + \delta_{k0}G \tag{2.12a}$$

$$\dot{v}_k = -\left[(u-c)\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right]_k - C_1^{-1}D_y C_2 p_k + \frac{1}{Re}(D_{xk}^2 + C_1^{-1}D_y^2 C_1)v_k \tag{2.12b}$$

$$0 = D_{xk}C_2^{-1}C_1 u_k + C_2^{-1}D_y C_1 v_k, \tag{2.12c}$$

for vectors $u_k = (u_{k1}, \ldots, u_{kM-1}), v_k = (v_{k1}, \ldots, v_{kM-1}), p_k = (p_{k0}, \ldots, p_{kM-1})$. The matrix D_{xk} is defined as $D_{xk}w_k = \mathrm{i}k\alpha w_k$. The dimension of the matrices is adjusted according to each particular case. For instance, in the term $D_{xk}C_2^{-1}C_1 u_k$, the matrices D_{xk}, C_2^{-1} and C_1 have respective dimensions $M \times M$, $M \times (M+1)$ and $(M+1) \times (M-1)$. We remark that we do not take advantage of fast Fourier transforms for linear terms, because the corresponding matrices are constant at each time step as we will see below.

2.3 Evaluation of nonlinear terms

One of the main difficulties in the application of a spectral Galerkin method is the evaluation of non-linear terms. In the following development we follow section 3.2 of Canuto *et al.*(1988) for the case of quadratic non-linearities (the ones that appear in the Navier–Stokes equations), so as to obtain an efficient algorithm for the evaluation of convolution sums. At this point we show some of the actual complexities in implementing the method. We have chosen the method of Galerkin–Fourier in x and collocation–Chebyshev in y, as opposed to Galerkin in both Fourier and Chebyshev, because the non-linear terms are much more awkward and expensive to evaluate in this latter case.

Let us describe how to evaluate convolution sums. We consider two truncated Fourier series

$$u(x) = \sum_{m=-N}^{N} \hat{u}_m \mathrm{e}^{\mathrm{i}m\alpha x}, \qquad v(x) = \sum_{n=-N}^{N} \hat{v}_n \mathrm{e}^{\mathrm{i}n\alpha x}, \tag{2.13}$$

which we extent up to order $P > N$ by defining $\hat{u}_k = \hat{v}_k = 0$, for $N < |k| \leqslant P$, and we want to calculate $w(x) = u(x)v(x)$, the product series truncated up to order N

$$w(x) = \sum_{k=-N}^{N} \hat{w}_k \mathrm{e}^{\mathrm{i}k\alpha x}, \qquad \hat{w}_k = \sum_{\substack{m+n=k \\ |m|,|n| \leqslant P}} \hat{u}_m \hat{v}_n, \tag{2.14}$$

where $\hat{w}_k$ is obtained by multiplying the series in (2.13) and grouping terms. This direct method for evaluating convolution sums requires $O(N^2)$ operations. From proposition B.2 we can consider the trigonometric interpolating polynomial of $w(x)$ at the points $\bar{x}_j = jL/(2P+1)$, for $j =$

$0, \ldots, 2P$, to approximate $\hat{w}_k$ by $\tilde{w}_k$, as for $|k| \leqslant N$,

$$\begin{aligned}
\tilde{w}_k &= \frac{1}{2P+1} \sum_{j=0}^{2P} w_j \mathrm{e}^{-\mathrm{i}k\alpha\bar{x}_j} = \frac{1}{2P+1} \sum_{j=0}^{2P} u_j v_j \mathrm{e}^{-\mathrm{i}k\alpha\bar{x}_j} \\
&= \frac{1}{2P+1} \sum_{j=0}^{2P} \left(\sum_{m=-P}^{P} \hat{u}_m \mathrm{e}^{\mathrm{i}m\alpha\bar{x}_j} \sum_{n=-P}^{P} \hat{v}_n \mathrm{e}^{\mathrm{i}n\alpha\bar{x}_j} \right) \mathrm{e}^{-\mathrm{i}k\alpha\bar{x}_j} \\
&= \frac{1}{2P+1} \sum_{m=-P}^{P} \sum_{n=-P}^{P} \hat{u}_m \hat{v}_n \sum_{j=0}^{2P} \mathrm{e}^{\mathrm{i}(m+n-k)\alpha\bar{x}_j} \\
&= \sum_{\substack{m+n=k \\ |m|,|n| \leqslant P}} \hat{u}_m \hat{v}_n + \sum_{\substack{m+n=k\pm(2P+1) \\ |m|,|n| \leqslant P}} \hat{u}_m \hat{v}_n
\end{aligned} \tag{2.15}$$

The last step is a consequence of (B.2) in proposition B.2 with $p = m + n - k$. It is now clear from these formulae that this second method for calculating convolution sums is not exact. The discrepancy term from $\hat{w}_k$ in (2.14) is called the aliasing error. Our task now is to find a condition on P, in order to cancel out the aliasing error. We use the property $\hat{u}_m = \hat{v}_n = 0$, for $N < |m|, |n| \leqslant P$, to choose P so that, for $|k| \leqslant N$

$$m + n = k \pm (2P+1) \quad \Longrightarrow \quad |m| > N \text{ or } |n| > N \quad \Longrightarrow \quad \hat{u}_m \hat{v}_n = 0,$$

and in this way we will have guaranteed that the aliasing error is canceled. For $|m|, |n| \leqslant N$ implies $|m + n| \leqslant 2N$, the condition for P is

$$k \pm (2P+1) > 2N \quad \text{or} \quad k \pm (2P+1) < -2N, \quad \text{for } |k| \leqslant N.$$

Thus the worst cases in k are

$$-N \pm (2P+1) > 2N \quad \text{or} \quad N \pm (2P+1) < -2N.$$

Both possibilities drive us to choose P such that

$$2P + 1 > 3N \quad \Longleftrightarrow \quad P \geqslant \frac{3}{2} N.$$

For $P = 3N/2$, with the help of fast Fourier transforms, the operation count of this procedure to evaluate convolution sums is $O(N \log_2 N)$, substantially better than $O(N^2)$ for (2.14).

Algorithm to compute non-linear terms. From the previously described method to calculate convolution sums, let us precise the steps to evaluate non-linear terms in (2.2), namely

$$\left[\widehat{(u-c)\frac{\partial u}{\partial x}} + v\frac{\partial u}{\partial y} \right]_k, \qquad \left[\widehat{(u-c)\frac{\partial v}{\partial x}} + v\frac{\partial v}{\partial y} \right]_k. \tag{2.16}$$

We start from values of the Fourier harmonics $\hat{u}_k$ and $\hat{v}_k$ at y_m defined in (2.4)–(2.5).

1) Evaluate $\partial u/\partial x$ and $\partial v/\partial x$ by means of the linear transforms $D_{xk}\hat{u}_k = \mathrm{i}k\alpha\hat{u}_k$, $D_{xk}\hat{v}_k = \mathrm{i}k\alpha\hat{v}_k$ for $k = 0, \ldots, N$.
2) Evaluate $\partial u/\partial y$ and $\partial v/\partial y$. We take the transform C_1 in (2.7) by means of fast cosine transform described in proposition B.3, then algorithm (2.9) to evaluate y-derivatives and finally another fast cosine transform to perform C_1^{-1}.
3) Pad with zeros the harmonics of $u, v, \partial u/\partial x, \partial u/\partial y, \partial v/\partial x$ and $\partial v/\partial y$ from order $N+1$ till $P \geqslant 3N/2$, at each y_m for $m = 1, \ldots, M-1$.
4) Use the inverse fast Fourier transform (IFFT) to transform $\hat{u}_k$, $\hat{v}_k$, $\partial\hat{u}_k/\partial x$, $\partial\hat{u}_k/\partial y$, $\partial\hat{u}_k/\partial x$ and $\partial\hat{v}_k/\partial y$ back to physical space, in order to get u, v, $\partial u/\partial x$, $\partial u/\partial y$, $\partial v/\partial x$ and $\partial v/\partial y$ at $(\bar{x}_j, y_m)$ for $j = 0, \ldots, 2P$, $m = 1, \ldots, M-1$.
5) At the points $(\bar{x}_j, y_m)$ for $j = 0, \ldots, 2P$, $m = 1, \ldots, M-1$, compute

$$(u-c)\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}, \qquad (u-c)\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}.$$

6) Take fast Fourier transforms (FFT) of the values from the last step at each y_m for $m = 1, \ldots, M-1$, to return to Fourier space and so finally obtain for $k = 0, \ldots, N$ the desired harmonics (2.16).

We remark that all the Fourier transforms employed in this algorithm, even to evaluate cosine transforms, are of type complex to real or vice versa, which cost roughly one half of a complex to complex Fourier transform.

2.4 Reduced equations. Temporal evolution

Up to now in system (1.7), we have discretized spatial derivatives to obtain system (2.12), in which only remain temporal derivatives in (2.12a) and (2.12b), together with the algebraic equation (2.12c) corresponding to divergence free condition. Therefore (2.12) is a system of differential-algebraic equations. To simplify the study of the dynamics of (2.12), we convert it to a system of ordinary differential equations through several algebraic manipulations. From here, the stability of equilibrium solutions will be determined by the eigenvalues of the linear part of the system. In passing we reduce the dimension of system (2.12) from $(2N+1)(3M-2)$ to $(2N+1)(M-2)+1$ in the final equations, i.e. roughly one third of the original dimension. In this section the study is made for the case of constant pressure gradient. In §2.5 we consider the case of constant flux.

Reduced equations. Our goal is to get rid of v and p in (2.12). We start from vectors of complex values $u_k = (u_{k1}, \ldots, u_{kM-1})^t$, $v_k = (v_{k1}, \ldots, v_{kM-1})^t$ and $p_k = (p_{k0}, \ldots, p_{kM-1})^t$ for $k = 0, \ldots, N$, corresponding to the Fourier coefficients of u, v and p' as defined in (2.5). In particular u_0, v_0 and p_0 are real vectors and the rest are complex. Likewise we define $\bar{u}_k = (u_{k1}, \ldots, u_{kM-2})^t$ and $\bar{v}_k = (u_{kM-1}, v_{k1}, \ldots, v_{kM-1})^t$ for $k = 1, \ldots, N$. With this split of variables, from (2.12c) $\bar{v}_k$ may solved from $\bar{u}_k$ and thus we can obtain a matrix T_k that carries out the transformation $\bar{v}_k = T_k\bar{u}_k$. The dimension of T_k is $M \times (M-2)$, the entries on its first row are real and on the rest purely imaginary, as can be verified directly. For $k = 0$, (2.12c) is written as $\partial v_0/\partial y = 0$ which together with (2.6), $v_0(\pm 1) = 0$, gives $v_0(y) = 0$ and therefore we set $v_{01} = \cdots = v_{0,M-1} = 0$. As a consequence, since in (2.12a) there are no pressure terms for $k = 0$, this is the only equation which we need to be consider and it only depends on u, once the substitution $\bar{v}_k = T_k\bar{u}_k$ is applied.

For $k = 1, \dots, N$ we introduce the notation

$$
\begin{aligned}
U_k &= -\left[(u-c)\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right]_k + \frac{1}{Re}(D_{xk}^2 + C_1^{-1}D_y^2 C_1)u_k + \delta_{k0}G \\
V_k &= -\left[(u-c)\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right]_k + \frac{1}{Re}(D_{xk}^2 + C_1^{-1}D_y^2 C_1)v_k, \\
\bar{U}_k &= (U_k)_{\{1,\dots,M-2\}}, \qquad \bar{Q}_k = (D_{xk}C_1^{-1}C_2)_{\{1,\dots,M-2\}}, \\
\bar{V}_k &= \begin{pmatrix} (U_k)_{\{M-1\}} \\ V_k \end{pmatrix}, \qquad Q_k = \begin{pmatrix} (D_{xk}C_1^{-1}C_2)_{\{M-1\}} \\ C_1^{-1}D_y C_2 \end{pmatrix},
\end{aligned}
$$

where $A_{\{i_1,\dots,i_n\}}$ stands for rows $i_1, \dots, i_n$ of matrix A. Equations (2.12a) and (2.12b) are now expressed as

$$
\begin{cases}
\dot{\bar{u}}_k = \bar{U}_k - \bar{Q}_k p_k \\
\dot{\bar{v}}_k = \bar{V}_k - Q_k p_k.
\end{cases}
$$

The matrix Q_k turns out to be an $M \times M$ invertible matrix, so from the second equation we obtain $p_k = Q_k^{-1}(\bar{V}_k - \dot{\bar{v}}_k)$, which substituting on the first yields

$$
\dot{\bar{u}}_k = \bar{U}_k - \bar{Q}_k Q_k^{-1}(\bar{V}_k - \dot{\bar{v}}_k) = \bar{U}_k - \bar{Q}_k Q_k^{-1}(\bar{V}_k - T_k \dot{\bar{u}}_k),
$$

and finally letting $P_k = \bar{Q}_k Q_k^{-1}$, it is also possible to invert $I - P_k T_k$, and thus we may solve for $\dot{\bar{u}}_k$

$$
\begin{cases}
\dot{u}_0 = U_0 \\
\dot{\bar{u}}_k = (I - P_k T_k)^{-1}(\bar{U}_k - P_k \bar{V}_k), \quad k = 1, \dots, N,
\end{cases}
\tag{2.17}
$$

where I is the identity matrix of dimension $M-2$ and we have extended the definition of U_k for $k = 0$. Bearing in mind the substitution $\bar{v}_k = T_k \bar{u}_k$, we observe that system (2.17) does not depend on $\bar{v}_k$ and p_k: it only depends on u_0 and $\bar{u}_k$ for $k = 1, \dots, N$. As was announced at the beginning of this section the real dimension of (2.17) is $(2N+1)(M-2)+1$. In addition, due to the elimination of pressure in (2.17), we avoid the indeterminacy caused by an additive constant, which has no effect on the pressure gradient.

Temporal evolution. Once removed v and p from (2.12), in (2.17) it just remains to discretize temporal derivatives. We have chosen a semi-implicit finite difference method, attending several factors as computational cost, stability, accuracy and storage requirements. The scheme adopted is typical for Navier–Stokes equations and employs the implicit Crank–Nicolson's method

$$
w^{n+1} = w^n + \frac{\Delta t}{2}\left[F(w^{n+1}) + F(w^n)\right], \tag{2.18}
$$

for diffusion (linear terms: pressure and viscosity), and the explicit Adams–Bashforth's method

$$
w^{n+1} = w^n + \frac{\Delta t}{2}\left[3F(w^n) - F(w^{n-1})\right], \tag{2.19}
$$

for advection (non-linear terms), being $w^n = w(t_n)$ for $t_n = n\Delta t$, and $\dot{w} = F(w)$ the ODE being approximated. To apply different methods on each term of F, we suppose $F(w) = \mathcal{L}(w) + \mathcal{N}(w)$ and thus we write $\dot{w} = F(w)$ in integral form as

$$\begin{aligned} w(t_{n+1}) &= w(t_n) + \int_{t_n}^{t_{n+1}} \mathcal{L}(w(t))\,\mathrm{d}t + \int_{t_n}^{t_{n+1}} \mathcal{N}(w(t))\,\mathrm{d}t \\ &\approx w^n + \frac{\Delta t}{2}\left[\mathcal{L}(w^{n+1}) + \mathcal{L}(w^n)\right] + \frac{\Delta t}{2}\left[3\mathcal{N}(w^n) - \mathcal{N}(w^{n-1})\right], \end{aligned}$$

where the approximation of the integrals is based on methods (2.18) and (2.19) respectively. Arranging terms, we take a step of the method by solving

$$w^{n+1} - \frac{\Delta t}{2}\mathcal{L}(w^{n+1}) = w^n + \frac{\Delta t}{2}\left[\mathcal{L}(w^n) + 3\mathcal{N}(w^n) - \mathcal{N}(w^{n-1})\right]. \tag{2.20}$$

Recurrence (2.20) is a two-step method, so it needs the solution at two consecutive time values t_{n-1}, t_n in order to get it at t_{n+1}. When (2.20) begins, w^0 comes from initial conditions and to generate w^1 we construct a one-step method. In this case linear and non-linear terms are discretized by means of implicit and explicit Euler's method respectively

$$w^{n+1} = w^n + \Delta t F(w^{n+1}), \qquad w^{n+1} = w^n + \Delta t F(w^n),$$

which, each time step of length $\Delta t/2$, yield the semi-implicit method

$$w^{n+1} - \frac{\Delta t}{2}\mathcal{L}(w^{n+1}) = w^n + \frac{\Delta t}{2}\mathcal{N}(w^n). \tag{2.21}$$

The error incurred in (2.18) and (2.19) with respect to the exact solution is $O((\Delta t)^2)$, as is straightforward to check. Crank–Nicolson's method is absolutely stable in the entire left-half plane, i.e. if $\mathrm{Re}(\lambda\Delta t) \leqslant 0$ then the approximated solution w^n of the scalar problem $\dot{w} = \lambda w$ is bounded as $n \to \infty$. Instead for Adams–Bashforth's method the stability region is an area of the complex plane included in $[-1, 0] \times [-1, 1]$. The stability in this case depends on the eigenvalues λ of the linear part of the spatial discretization in (2.12) and Δt must be restricted according to $\mathrm{Re}(\lambda\Delta t) \leqslant 0$ so as to get stability. As we will see in §2.6, for the kind of solutions considered in our study, the time step Δt employed is in the stability region, since the estimated errors are kept small.

Let us see how the actual implementation is tackled. We can express (2.17) as

$$\dot{\bar{u}}_k = \mathcal{L}_k(\bar{u}_k) + \mathcal{N}_k(\bar{u}_0, \ldots, \bar{u}_N), \quad k = 0, \ldots, N, \tag{2.22}$$

where $\bar{u}_0 = u_0$ and $\mathcal{L}_k$, $\mathcal{N}_k$ corresponds respectively to linear and nonlinear terms in $\bar{u}_0, \ldots, \bar{u}_N$ on the right hand side of (2.17). At this point we consider only the case when a constant pressure gradient G is held constant and put Re_p as the corresponding Reynolds number. The equations for constant flux are studied in §2.5. To emphasize the dependence on each of the variables in (2.22),

we precise their formulae:

$$
\begin{aligned}
Re_p\mathcal{L}_0(\bar{u}_0) &= C_1^{-1}D_y^2C_1\bar{u}_0,\\
Re_p(I-P_kT_k)\mathcal{L}_k(\bar{u}_k) &= (D_{xk}^2+C_1^{-1}D_y^2C_1)_{\{1,\dots,M-2\}}\begin{pmatrix} I \\ (T_k)_{\{1\}}\end{pmatrix}\\
&\quad - P_k\begin{pmatrix}(D_{xk}^2+C_1^{-1}D_y^2C_1)_{\{M-1\}} & 0\\ 0 & D_{xk}^2+C_1^{-1}D_y^2C_1\end{pmatrix}\begin{pmatrix} I\\ T_k\end{pmatrix}\bar{u}_k,\\
\mathcal{N}_0(\bar{u}_0,\dots,\bar{u}_N) &= -\left[(u-c)\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right]_0+G,\\
(I-P_kT_k)\mathcal{N}_k(\bar{u}_0,\dots,\bar{u}_N) &= -\left(\left[(u-c)\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right]_k\right)_{\{1,\dots,M-2\}}\\
&\quad + P_k\begin{pmatrix}\left(\left[(u-c)\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right]_k\right)_{\{M-1\}}\\ \left[(u-c)\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}\right]_k\end{pmatrix},
\end{aligned}
\tag{2.23}
$$

for $k=1,\dots,N$. It is easy to check that $P_k=\bar{Q}_kQ_k^{-1}$ has its first column real and the remaining ones purely imaginary. Then P_kT_k is a real matrix and thus the matrix of $\mathcal{L}_k$ for $k=0,\dots,N$ is also real. The substitution of (2.22) in (2.20) gives for $k=0,\dots,N$

$$
\bar{u}_k^{n+1}-\frac{\Delta t}{2}\mathcal{L}_k(\bar{u}_k^{n+1})=\bar{u}_k^n+\frac{\Delta t}{2}\left[\mathcal{L}_k(\bar{u}_k^n)+3\mathcal{N}_k^n-\mathcal{N}_k^{n-1}\right],
$$

which can be abbreviated as

$$
A_k\bar{u}_k^{n+1}=b_k^{n+1}, \tag{2.24}
$$

where

$$
A_k=I-\frac{\Delta t}{2}\mathcal{L}_k,\qquad b_k^{n+1}=\bar{u}_k^n+\frac{\Delta t}{2}\left[\mathcal{L}_k(\bar{u}_k^n)+3\mathcal{N}_k^n-\mathcal{N}_k^{n-1}\right],
$$

and $\mathcal{N}_k^j=\mathcal{N}_k(\bar{u}_0^j,\dots,\bar{u}_N^j)$. For $k=0$ the identity matrix I has dimension $M-1$. We remark that A_k is a real matrix of size $M-1$ for $k=0$ and $M-2$ for $k=1,\dots,N$ and that it only depends on M and Δt which are kept constant on each flow simulation. Thus we simply have to compute once the LU decomposition of A_k. Recurrence (2.24) needs $\bar{u}_0,\dots,\bar{u}_N$ at two time instants. The first one is taken from initial conditions and for the second one, adapting (2.21) to our case, we have

$$
\bar{u}_k^{n+1}-\frac{\Delta t}{2}\mathcal{L}_k(\bar{u}_k^{n+1})=\bar{u}_k^n+\frac{\Delta t}{2}\mathcal{N}_k^n, \tag{2.25}
$$

which must be applied twice in order to get the solution $\bar{u}_k^1$ at $t=\Delta t$. It is not a coincidence that A_k is also the matrix of the system to be solved in (2.25). In this way we can take advantage of the same LU decomposition and the corresponding storage.

Time marching scheme. We start from fixed values of Δt, M, N, $K=(2N+1)(M-2)+1$ and $(\bar{u}_0^0,\dots,\bar{u}_N^0)\in\mathbb{R}^K$ at the time instant $t=0$. In the following algorithm for the time evolution, all the steps refer to $k=0,\dots,N$. It is based on (2.24) and (2.25):

1) Calculate matrices A_k together with their LU decomposition.
2) Evaluate $b_k^{1/2} = \bar{u}_k^0 + \Delta t \mathcal{N}_k^0/2$ and solve $A_k \bar{u}_k^{1/2} = b_k^{1/2}$ for $\bar{u}_k^{1/2}$.
3) Evaluate $b_k^1 = \bar{u}_k^{1/2} + \Delta t \mathcal{N}_k^{1/2}/2$ and solve $A_k \bar{u}_k^1 = b_k^1$ for $\bar{u}_k^1$.
4) For $n = 1, 2, 3 \ldots$ obtain $b_k^{n+1} = 2\bar{u}_k^n - b_k^n + \Delta t \big[3\mathcal{N}_k^n - \mathcal{N}_k^{n-1}\big]/2$ and solve $A_k \bar{u}_k^{n+1} = b_k^{n+1}$ for $\bar{u}_k^{n+1}$.

To sum up, each time step the computational cost consist of evaluating $\mathcal{N}_k^n$, solving a linear system of size $M-1$ and $2N$ linear systems of size $M-2$, whose LU decomposition is already computed. The main storage requirements are the LU decompositions of matrices A_k, i.e. $(M-1)^2 + 2N(M-2)^2$ real coefficients. This has been an important reason to choose the numerical discretization as Fourier–Galerkin in x and Chebyshev–collocation in y (see § 2.1), instead of Chebyshev–Galerkin in y and Fourier–collocation in x. With this latter approach the linear terms are coupled in one whole matrix, as contrasted with one block of size $M-1$ and $2N$ of size $M-2$ in the implemented method as we have seen in (2.24).

It is also worth to mention the use of some library routines in the implementation of (2.24). Fast Fourier transforms have been calculated by means of the library functions FFTW (Fast Fourier Transform in the West, `http://www.fftw.org`), which authors claim to be usually faster than all other freely-available Fourier transform programs found on the Net. From the authors' manual: *FFTW is unique in that it automatically adapts itself to your machine, your cache, the size of your memory, the number of registers, and all the other factors that normally make it impossible to optimize a program for more than one machine.* To implement operations which imply vectors and matrices we have chosen ATLAS (Automatically Tuned Linear Algebra Software, `http://math-atlas.sourceforge.net`), which are also adapted routines to the specific architecture where the code is going to be exploited. Both sets of library functions have meant a worthy increase in the performance of the numerical integrator (2.24).

2.5 The constant flux numerical integrator

We have to make little changes in equations of § 2.4 to implement a numerical integrator which keeps the flux Q constant in time. According to table 1.2 we have to impose $Q = 4/3$ and from (1.9) applied to the non-dimensional case it turns out

$$\begin{aligned} G &= \frac{1}{2}\int_{-1}^{1} \frac{\partial \hat{u}_0}{\partial t}\,\mathrm{d}y - \frac{1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1} = \frac{1}{2}\frac{\partial}{\partial t}\int_{-1}^{1} \hat{u}_0\,\mathrm{d}y - \frac{1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1} \\ &= \frac{1}{2}\frac{\partial Q}{\partial t} - \frac{1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1} = -\frac{1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1}, \end{aligned} \tag{2.26}$$

for $\hat{u}_0$ as was defined in (2.1). Therefore, from the last expression, the restriction in G has implicit a constant flux, as we have used $\partial Q/\partial t = 0$ in its obtaining. Nevertheless, numerically (2.26) is not enough to keep the flux constant because, due to rounding errors, the flux is slightly varied each time step, producing substantial errors in long time integrations. Hence, in addition to imposing (2.26) as the mean pressure gradient, we also restrict the solution to $Q = 4/3$ each time instant. Both restrictions affect mainly the equation for $u_0 = (u_{01}, \ldots, u_{0M-1})$ in (2.17), because Q and G depend only on $\hat{u}_0$ and the dependence is linear. We incorporate them to $\mathcal{L}_0(u_0)$ of (2.23).

Calculation of G. By means of the transformations C_1 and D_y defined in (2.7) and (2.11) respectively, the linear operation $u_0' = (u_{00}', \ldots, u_{0M-1}') = D_y C_1 u_0$ computes the coefficients of the Chebyshev polynomial $\partial \hat{u}_0/\partial y = \sum_{m=0}^{M-1} u_{0m}' T_m(y)$. From it we obtain

$$\begin{aligned} G &= \frac{-1}{2Re_Q}\left[\frac{\partial \hat{u}_0}{\partial y}\right]_{-1}^{1} = \frac{-1}{2Re_Q}\sum_{m=0}^{M-1} u_{0m}'(T_m(1) - T_m(-1)) \\ &= \frac{-1}{2Re_Q}\sum_{m=0}^{M-1} u_{0m}'(\cos m0 - \cos m\pi) = \frac{-1}{2Re_Q}\sum_{m=0}^{M-1} u_{0m}'(1-(-1)^m) = \frac{-1}{Re_Q}\sum_{\substack{m=1\\ m \text{ odd}}}^{M-1} u_{0m}', \end{aligned}$$

and thus we modify the linear terms adding G

$$Re_Q \mathcal{L}_0(u_0) = C_1^{-1} D_y^2 C_1 u_0 + G = (C_1^{-1} D_y^2 C_1 - O D_y C_1) u_0,$$

where $O = (o_{ij})$ is a $(M-1) \times M$ matrix with $o_{ij} = 1$ if j odd and $o_{ij} = 0$ otherwise.

Calculation of Q. We put $\tilde{u}_0 = C_1 u_0$ and as we have seen in (1.8)

$$\begin{aligned} Q &= \int_{-1}^{1} \hat{u}_0(y)\,\mathrm{d}y = \int_{-1}^{1} \sum_{m=0}^{M} \tilde{u}_{0m} T_m(y)\,\mathrm{d}y = \sum_{m=0}^{M} \tilde{u}_{0m} \int_{-1}^{1} \cos(m \arccos(y))\,\mathrm{d}y \\ &= \sum_{m=0}^{M} \tilde{u}_{0m} \int_0^{\pi} \cos(m\theta)\sin\theta\,\mathrm{d}\theta = \sum_{\substack{m=0\\ m \text{ even}}}^{M} \frac{2\tilde{u}_{0m}}{1-m^2} = q^t u_0, \end{aligned} \tag{2.27}$$

where $q = (q_1, \ldots, q_{M-1})^t = (q_0', \ldots, q_M')C_1$, for $q_m' = 2/(1-m^2)$ if m even and $q_m' = 0$ otherwise. Now the condition $Q = 4/3$ is transformed to $q^t u_0 = 4/3$, which taking M even can be solved for $u_{0M/2}$ by

$$u_{0M/2} = \frac{1}{q_{M/2}}\left(\frac{4}{3} - \bar{q}^t \bar{u}_0\right), \tag{2.28}$$

where $\bar{q}$ and $\bar{u}_0$ represent vectors q and u_0 without the $M/2$–th component. Putting $(\mathcal{L}_0)^{\{M/2\}}$ as the $M/2$–th column of $\mathcal{L}_0$, and $\tilde{\mathcal{L}}_0$ as $\mathcal{L}_0$ without $(\mathcal{L}_0)^{\{M/2\}}$, then we may eliminate $u_{0M/2}$ from $\mathcal{L}_0$ since

$$\begin{aligned} \mathcal{L}_0(u_0) &= \tilde{\mathcal{L}}_0(\bar{u}_0) + (\mathcal{L}_0)^{\{M/2\}} u_{0M/2} \\ &= \left(\tilde{\mathcal{L}}_0 - \frac{1}{q_{M/2}}(\mathcal{L}_0)^{\{M/2\}}\bar{q}^t\right)(\bar{u}_0) + \frac{4}{3q_{M/2}}(\mathcal{L}_0)^{\{M/2\}}. \end{aligned}$$

Finally letting $\bar{\mathcal{L}}_0$ be $\tilde{\mathcal{L}}_0 - (\mathcal{L}_0)^{\{M/2\}}\bar{q}^t/q_{M/2}$ but its $M/2$–th row, the equation for $\bar{u}_0$ is

$$\dot{\bar{u}}_0 = \bar{\mathcal{L}}_0(\bar{u}_0) + \mathcal{N}_0(\bar{u}_0, \ldots, \bar{u}_N),$$

where

$$\mathcal{N}_0(\bar{u}_0, \ldots, \bar{u}_N) = -\left[(u-c)\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right]_0 + \frac{4}{3q_{M/2}}(\mathcal{L}_0)^{\{M/2\}}.$$

The dimension of the system is reduced by one with respect to (2.17). Equations for $\bar{u}_1, \ldots, \bar{u}_N$ have linear terms as in (2.17). In the evaluation of convective terms $u_{0M/2}$ is substituted from (2.28). The temporal evolution is implemented as in (2.24).

2.6 Check of the numerical integrator

In this section we verify that local errors originated in (2.24) from the time discretization, behave reasonably for the kind of solutions considered in this work and moderate values of Re. To that purpose, we approximate temporal derivatives by central finite differences and then we extrapolate those approximations.

Extrapolation method. Let us consider a given an expansion in powers of h, evaluated at h and λh

$$T(h) = \tau_0 + \tau_1 h^r + O(h^{r+1}), \qquad T(\lambda h) = \tau_0 + \tau_1 \lambda^r h^r + O(h^{r+1}).$$

We combine both expansion to cancel out terms of order r by

$$\frac{\lambda^r}{1-\lambda^r}\left(\frac{T(\lambda h)}{\lambda^r} - T(h)\right) = \tau_0 + O(h^{r+1}). \tag{2.29}$$

To approximate derivatives we use the central difference formula

$$D(h) = \frac{f(t+h) - f(t-h)}{2h}, \tag{2.30}$$

which from Taylor's expansions we know that

$$D(h) = a_0 + a_1 h^2 + \ldots + a_m h^{2m} + h^{2m+2}\beta_{m+1}(h),$$

where $a_{2k+1} = f^{2k+1}(t)/(2k+1)!$ for $k = 0, 1, \ldots, m+1$ and $\beta_{m+1}(h) \to a_{m+1}$ as $h \to 0$. Putting $\lambda = 1/2$ and $r = 2$, the extrapolation method applied to $D(h)$ is written as

$$D(h/2) + \frac{D(h/2) - D(h)}{3} = f'(t) + O(h^4). \tag{2.31}$$

Effective formula for the error in time. We denote $u^n = u(n\Delta t)$ for $n = 0, 1, \ldots$ and $u = (\bar{u}_0, \ldots, \bar{u}_N)$ in its discrete form as defined in §2.4. The procedure adopted to estimate the errors committed in the temporal evolution considers five consecutive instants of u, namely u^{n-2}, u^{n-1}, u^n, u^{n+1}, and u^{n+2}. Using (2.30) for $h = 2\Delta t$ and $h = \Delta t$, we approximate respectively $\dot{u}^n$ by

$$\frac{u^{n+2} - u^{n-2}}{4\Delta t} = \dot{u}^n + O((\Delta t)^2), \qquad \frac{u^{n+1} - u^{n-1}}{2\Delta t} = \dot{u}^n + O((\Delta t)^2),$$

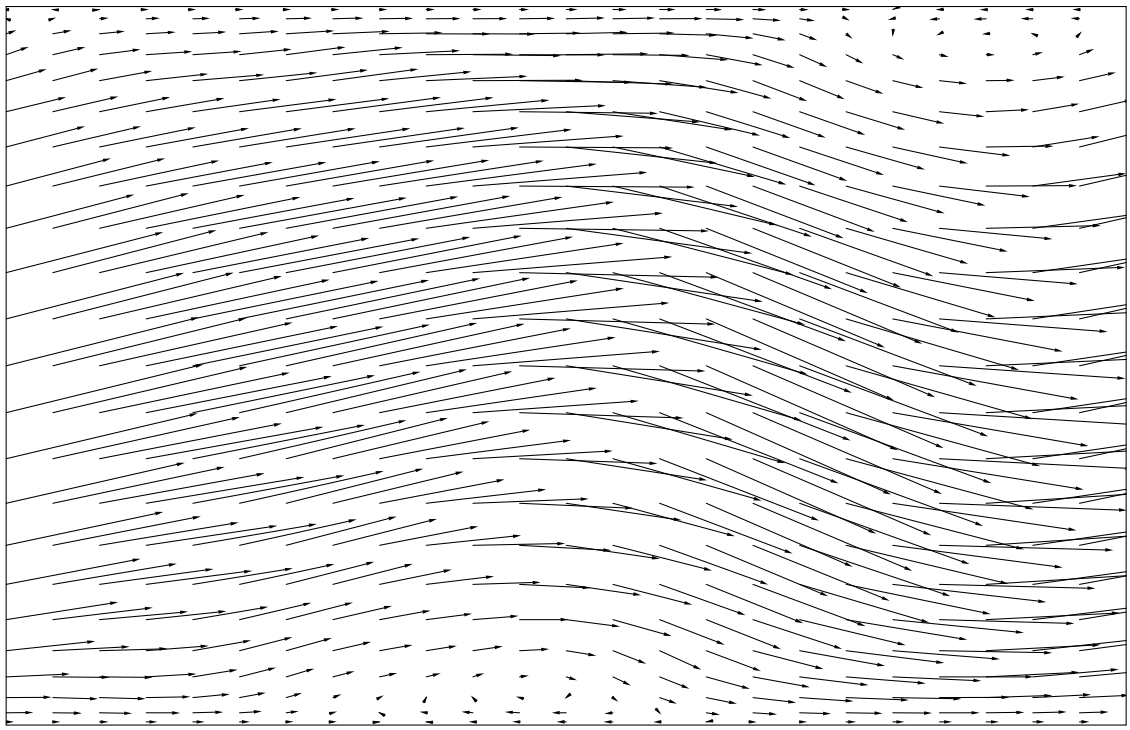
which combined through (2.31) yield

$$\frac{u^{n+1} - u^{n-1}}{2\Delta t} + \frac{1}{3}\left(\frac{u^{n+1} - u^{n-1}}{2\Delta t} - \frac{u^{n+2} - u^{n-2}}{4\Delta t}\right) = \dot{u}^n + O((\Delta t)^4). \tag{2.32}$$

We condense (2.22) as $\dot{u} = \mathcal{L}(u) + \mathcal{N}(u)$, putting $\mathcal{L} = (\mathcal{L}_0, \ldots, \mathcal{L}_N)$ and $\mathcal{N} = (\mathcal{N}_0, \ldots, \mathcal{N}_N)$. On the other hand, (2.24) can be transformed in

$$\mathcal{L}(u^n) = \frac{2}{\Delta t}(u^n - b^n),$$

a



b

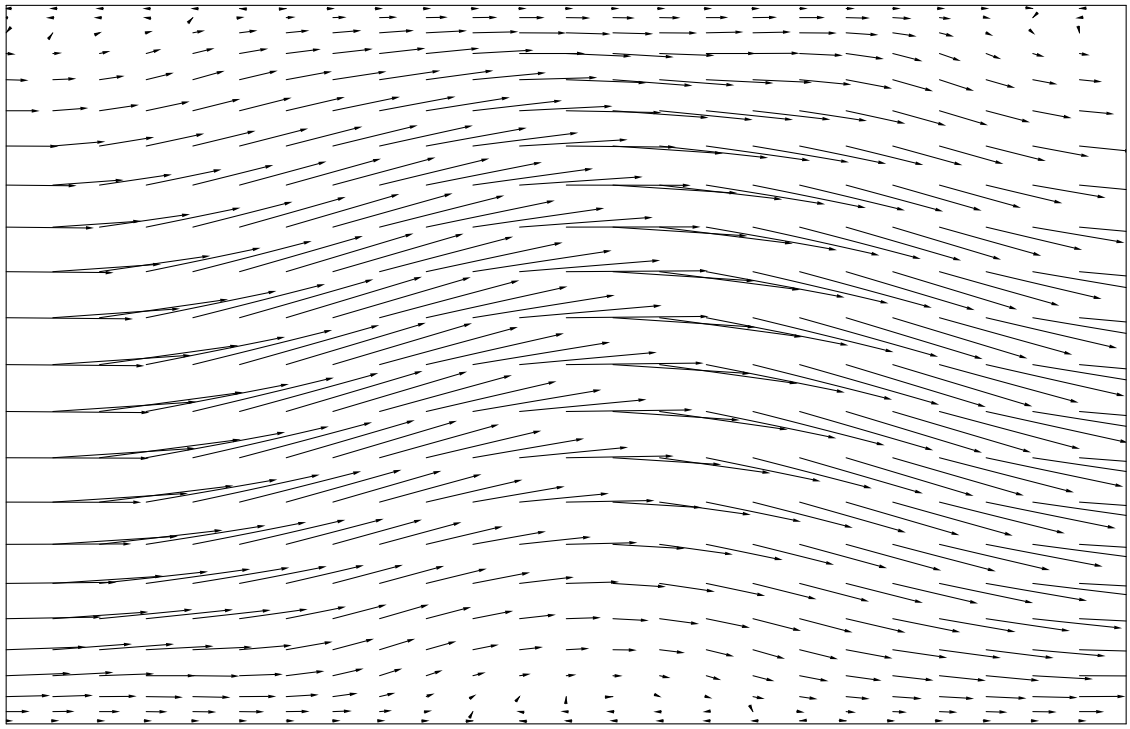


FIGURE 2.1. Field of velocities (u, v) for one time instant of fbws marked as *a*, *b* in table 2.3. The represented frame corresponds to $[0, L] \times [-1, 1]$.

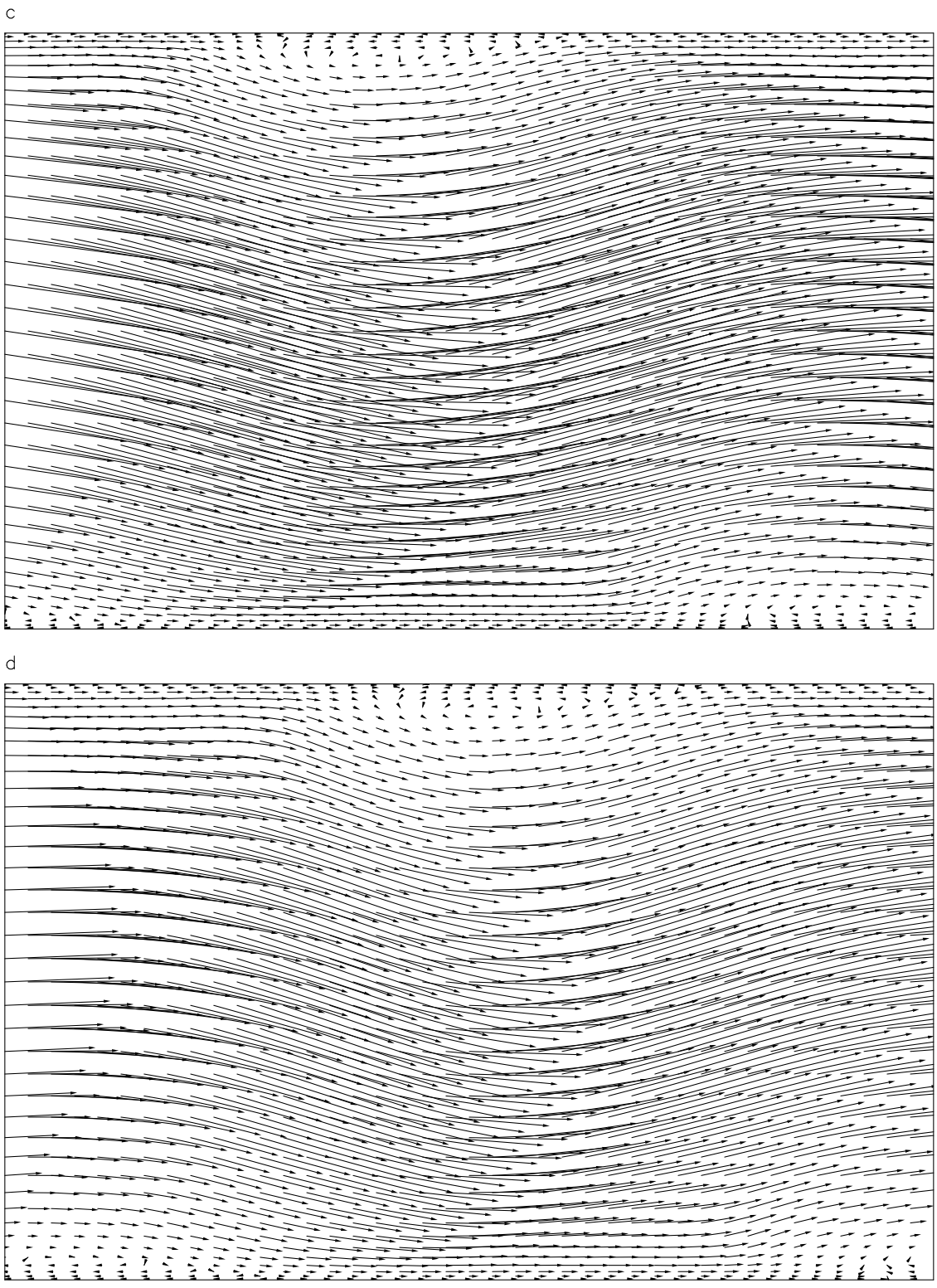


FIGURE 2.2. Analogous of fi gure 2.1 for fbws marked as *c*, *d* in table 2.3.

where $b = (b_0, \dots, b_N)$. Because (2.18) and (2.19) produce $O((\Delta t)^2)$ errors, this is also so for (2.24). Consequently the time evolution using (2.24) yields $\dot{u}^n = \mathcal{L}(u^n) + \mathcal{N}(u^n) + O((\Delta t)^2)$, which substituted in (2.32) gives a final expression for the error

$$\frac{1}{12\Delta t}\left(u^{n-2} - 8u^{n-1} + 24b^n - 24u^n - 12\Delta t\mathcal{N}^n + 8u^{n+1} - u^{n+2}\right) = O((\Delta t)^2). \quad (2.33)$$

Norm of a flow. To measure the size of a flow, and in particular the expression in (2.33), we need a norm. Given a profile of velocities $(u, v)(x, y)$, based on the L_2-norm, we define its norm $\|(u, v)\|$ as

$$\|(u, v)\|^2 \stackrel{\text{def}}{=} \frac{1}{L}\int_0^L \int_{-1}^1 \left[u(x, y)^2 + v(x, y)^2\right] \mathrm{d}y\, \mathrm{d}x. \quad (2.34)$$

In order to evaluate (2.34) for a discretized flow (u, v) as in (2.1) and (2.5), if we put $w(x, y) = u(x, y)^2 + v(x, y)^2$, expressed as Fourier series $w(x, y) = \sum_{k\in\mathbb{Z}} w_k(y)\mathrm{e}^{\mathrm{i}k\alpha x}$, then

$$\begin{aligned}\|(u, v)\|^2 &= \frac{1}{L}\int_0^L \int_{-1}^1 w(x, y)\, \mathrm{d}y\, \mathrm{d}x = \int_{-1}^1 \sum_{k\in\mathbb{Z}} \frac{w_k(y)}{L} \int_0^L \mathrm{e}^{\mathrm{i}k\alpha x}\, \mathrm{d}x\, \mathrm{d}y \\ &= \int_{-1}^1 w_0(y)\, \mathrm{d}y = q^t w_0.\end{aligned} \quad (2.35)$$

The last step is a direct consequence of (2.27), where $w_0 = (w_{01}, \dots, w_{0M-1})$, corresponding to notation (2.5). Moreover, for the truncated series of $w(x, y)$, from the definition of convolution sums in (2.14), we obtain for $m = 1, \dots, M - 1$

$$w_{0m} = \sum_{k=-N}^{N} (u_{km}u_{-km} + v_{km}v_{-km}) = u_{0m}^2 + 2\sum_{k=1}^{N}(|u_{km}|^2 + |v_{km}|^2),$$

which finally allows us to evaluate (2.35).

In order to evaluate the error formula in (2.33), we first need to apply the transforms T_k defined in §2.4 to compute the components of u, v not present in $(u_0, \bar{u}_1, \dots, \bar{u}_N)$, and then we can apply (2.35). In table 2.3 we present errors, according to (2.33), for different flows. We observe that for fixed values of Re and $N \times M$, errors depends on $(\Delta t)^2$ as, when Δt is halved, they are roughly divided by 4. This is in agreement with (2.33). Errors are increased with Re and slightly with $N \times M$. Data in table 2.3 give only a reference of the precision of the numerical integrators, because errors depends strongly on the type solution being integrated: in this case it is about quasi-periodic flows which is the subject of chapter 4.

In figures 2.1 and 2.2 we plot vectors $(u, v)(x_j, y_m)$ for $x_j = jL/M$, $j = 0, 1, \dots, M - 1$ and y_m as defined in (2.4). From (2.1), (2.5) and because u, v are real functions, their obtaining at (x_j, y_m) is accomplished by

$$\begin{aligned}u(x_j, y_m) &= u_{0m} + 2\sum_{k=1}^{N} \mathrm{Re}(u_{km}\mathrm{e}^{\mathrm{i}k\alpha x_j}) = u_{0m} + 2\sum_{k=1}^{N} \left[u_{km}^r \cos(k\alpha x_j) - u_{km}^{\mathrm{i}} \sin(k\alpha x_j)\right] \\ v(x_j, y_m) &= 2\sum_{k=1}^{N} \mathrm{Re}(v_{km}\mathrm{e}^{\mathrm{i}k\alpha x_j}) = 2\sum_{k=1}^{N} \left[v_{km}^r \cos(k\alpha x_j) - v_{km}^{\mathrm{i}} \sin(k\alpha x_j)\right],\end{aligned}$$

where $\mathrm{Re}\, z = z^r$ represent both the real part of z, and $\mathrm{Im}\, z = z^{\mathrm{i}}$ its imaginary part. Again in this case we have employed transforms T_k.

Constant flux				Constant pressure gradient			
$N \times M$	Re_Q	Δt	error	$N \times M$	Re_p	Δt	error
4×24	5737.26	0.020	8.50×10^{-9}	4×24	7638.23^b	0.020	2.38×10^{-9}
4×24	5737.26	0.010	2.12×10^{-9}	4×24	7638.23	0.010	5.95×10^{-10}
4×24	5737.26	0.005	5.30×10^{-10}	4×24	7638.23	0.005	1.49×10^{-10}
4×24	6000.00	0.020	1.62×10^{-8}	4×24	8500.40	0.020	4.68×10^{-9}
4×24	6000.00	0.010	4.04×10^{-9}	4×24	8500.40	0.010	1.17×10^{-9}
4×24	6000.00	0.005	1.01×10^{-9}	4×24	8500.40	0.005	2.93×10^{-10}
4×24	7401.06^a	0.020	9.46×10^{-6}	4×24	9504.20	0.020	2.64×10^{-7}
4×24	7401.06	0.010	2.37×10^{-6}	4×24	9504.20	0.010	6.62×10^{-8}
4×24	7401.06	0.005	5.88×10^{-7}	4×24	9504.20	0.005	1.66×10^{-8}
7×40	5269.03	0.020	6.81×10^{-8}	7×40	6699.62	0.020	3.11×10^{-8}
7×40	5269.03	0.010	1.70×10^{-8}	7×40	6699.62	0.010	7.82×10^{-9}
7×40	5269.03	0.005	4.26×10^{-9}	7×40	6699.62	0.005	1.96×10^{-9}
7×40	5835.42^c	0.020	8.06×10^{-7}	7×40	7589.07	0.020	2.12×10^{-7}
7×40	5835.42	0.010	2.02×10^{-7}	7×40	7589.07	0.010	5.31×10^{-8}
7×40	5835.42	0.005	5.04×10^{-8}	7×40	7589.07	0.005	1.33×10^{-8}
7×40	6658.21	0.020	3.38×10^{-6}	7×40	8260.39	0.020	3.86×10^{-7}
7×40	6658.21	0.010	8.46×10^{-7}	7×40	8260.39	0.010	9.65×10^{-8}
7×40	6658.21	0.005	2.12×10^{-7}	7×40	8260.39	0.005	2.41×10^{-8}
7×40	7539.27	0.020	8.03×10^{-6}	7×40	9051.24^d	0.020	6.26×10^{-7}
7×40	7539.27	0.010	2.01×10^{-6}	7×40	9051.24	0.010	1.57×10^{-7}
7×40	7539.27	0.005	5.03×10^{-7}	7×40	9051.24	0.005	3.91×10^{-8}

TABLE 2.3. Errors, according to (2.33)–(2.35), committed during the integration of different flows for the specified values of Re_Q, Re_p, $N \times M$, Δt and for fixed $\alpha = 1.02056$. The error is computed as an average of several measurements of (2.33). For fixed values of Re and $N \times M$ the same flow is integrated for values of $\Delta t = 0.02, 0.01, 0.005$. Reynolds numbers marked with a superindex are also plotted in the corresponding graph of figures 2.1 and 2.2.

2.7 Poincaré sections

In order to study periodic and quasi-periodic orbits on time, we define a Poincaré section, Σ_1, of the fluid in terms of discrete variables defined in (2.5) and §2.4. This is a fundamental tool in analysing the behaviour of the fluid, so we need an accurate calculation to know the solution on Σ_1. In this section we detail the algorithm implemented to carry out this calculation. The actual definition of the Poincaré section is

$$\Sigma_1 = \{U = (\bar{u}_0, \ldots, \bar{u}_N) \mid S = 0\},$$

where $S = \mathrm{Re}(\bar{u}_{11}) - s_1$, for $s_1 \in \mathbb{R}$, a fixed value which we adapt according to the fluid being integrated. We are interested in finding $u(\bar{t}) \in \Sigma_1$, only for $\bar{t}$ such that $S(\bar{t}) = 0$ and $\partial S/\partial t(\bar{t}) > 0$, i.e. we only search a crossing through 0 of S if at the crossing point S is growing.

Algorithm to approach the Poincaré section. Let us suppose we are following the evolution of the fluid by means of algorithm (2.24) using a time step Δt. We denote by $t_n = n\Delta t$ and $S^n = S(t_n)$ for $n = 0, 1, 2, \ldots$. If for some $n \in \mathbb{N}$ we detect $S^n < 0$ and $S^{n+1} > 0$, then:

1) Using a smaller time step as $(\Delta t)^2$ and $u(t_n)$ as the initial condition, obtain $S(t)$ from two applications of algorithm (2.25) and successive of (2.24) until $S(\bar{t}) > 0$ for some $\bar{t} > t_n$. In this way we stay closer to Σ_1.
2) By means of Newton's method applied to $S(t) = 0$, select a new time step $\overline{\Delta t} = -S(\bar{t})/S'(\bar{t})$. The derivative $S'(\bar{t}) = \mathrm{Re}(\dot{u}_{11})$ is obtained from the right hand side of (2.22).
3) Apply one iteration of (2.25) with $\overline{\Delta t}$ computed in 2) and return to step 2) while $|S(t)| > \varepsilon$ where $\varepsilon > 0$ is some fixed tolerance.

For $\varepsilon = 10^{-11}$ we usually need from 1 to 3 Newton's iterations to reach $|S| \leqslant \varepsilon$.

2.8 Pseudo-arclength continuation method

In this section we treat basic ideas on continuation methods needed to traverse bifurcating curves of periodic and quasi-periodic solutions. They are an extract of chapters 1–9 of Allgower & Georg (1990).

Basic ideas. We consider $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$ a smooth mapping for which we want to find a curve of points $\gamma \in \mathbb{R}^{n+1}$ such that $H(\gamma) = 0$. If we know γ_0 such that $H(\gamma_0) = 0$ and $\mathrm{rank}(DH(\gamma_0)) = n$, then by the implicit function theorem there exists an open interval, $J \ni 0$ and a smooth curve $\zeta \in J \mapsto \gamma(\zeta) \in \mathbb{R}^{n+1}$ such that for all $\zeta \in J$: a) $\gamma(0) = \gamma_0$, b) $H(\gamma(\zeta)) = 0$, c) $DH(\gamma(\zeta)) \in \mathcal{R}_n = \{B \in \mathcal{M}_{n\times(n+1)} \mid \mathrm{rank}(B) = n\}$ and d) $\gamma'(\zeta) \neq 0$.

By derivating $H(\gamma(\zeta)) = 0$, the tangent vector to the curve, $\gamma'(\zeta)$, satisfies $DH(\gamma(\zeta))\gamma'(\zeta) = 0$ and thus $\gamma'(\zeta)$ is orthogonal to all rows of $DH(\gamma(\zeta))$, or equivalently the $(n+1) \times (n+1)$ matrix

$$C(\zeta) = \begin{pmatrix} DH(\gamma(\zeta)) \\ \gamma'(\zeta)^t \end{pmatrix}$$

is nonsingular for all $\zeta \in J$. We introduce the parameter arclength s by $s(\zeta) = \int_0^\zeta \|\gamma'(t)\|\,\mathrm{d}t$. For this new parameter s the tangent vector satisfies $\|\,\mathrm{d}\gamma/\,\mathrm{d}s(s)\| = \|\dot{\gamma}(s)\| = 1$, for $s \in I$ and I some new interval. We call the orientation of the curve positive when $\det C(s) > 0$ for $s \in I$.

Definition 2.2. *Let $A \in \mathcal{R}_n$. The unique vector $t(A) \in \mathbb{R}^{n+1}$ which satisfies the conditions:*

$$At = 0, \qquad \|t\| = 1, \qquad \det \begin{pmatrix} A \\ t^t \end{pmatrix} > 0,$$

is called the tangent vector induced by A.

Again using the implicit function theorem it is easy to prove that the map $A \in \mathcal{R}_n \mapsto t(A)$ has open domain and is smooth. This fact allows us to transform the problem of finding the positively oriented curve $\gamma(s) \in H^{-1}(0)$, into the initial value problem

$$\dot{\gamma} = t(DH(\gamma)), \qquad \gamma(0) = \gamma_0, \tag{2.36}$$

which suggest the application of numerical methods for solving this kind of problems. This will be the first step to predict an approximate solution. For instance we can apply the Euler method to (2.36): from a known point γ_i on the curve, we estimate γ_{i+1} by

$$\gamma_{i+1} = \gamma_i + ht(DH(\gamma_i)), \tag{2.37}$$

where $h > 0$ represents the step size, whose choice is described below. This first approximation of γ_{i+1} is what we call a predictor step. To improve its precision on the curve $H^{-1}(0)$, we use a Newton-like method detailed in what follows.

Obtaining of tangent vector and Moore–Penrose inverse. Since the Jacobian matrix $DH \in \mathcal{M}_{n\times(n+1)}$ is not square, in order to apply Newton's method to $H(\gamma) = 0$ we need to introduce a special right inverse of DH.

Definition 2.3. *Let $A \in \mathcal{R}_n$ (this implies that AA^t is nonsingular). Then the Moore–Penrose inverse of A is defined by $A^+ = A^t(AA^t)^{-1}$.*

The proof of the following lemmas can be found in Allgower & Georg (1990) p. 19.

Lemma 2.4. *Let $A \in \mathcal{R}_n$. Then, for all $b \in \mathbb{R}^n$ and $x \in \mathbb{R}^{n+1}$, the following statements are equivalent: a) $Ax = b$, $t(A)^t x = 0$; b) $x = A^+ b$; c) x solves the problem: $\min_w\{\|w\| \mid Aw = b\}$.*

Lemma 2.5. *If $A \in \mathcal{R}_n$ then: a) A^+A is the orthogonal projection from $\mathbb{R}^{n+1}$ onto $t(A)^\perp =$ range(A^t), i.e. $A^+A = I - t(A)t(A)^t$; b) $AA^+ = I$; c) If B is any right inverse of A, then $A^+ = (I - t(A)t(A)^t)B$.*

We consider $A \in \mathcal{R}_n$ for which we have calculated an LU decomposition of the form

$$PA^t = L\begin{pmatrix} U \\ 0^t \end{pmatrix},$$

where $L \in \mathcal{M}_{(n+1)\times(n+1)}$ is lower triangular, $U \in \mathcal{M}_{n\times n}$ is upper triangular and $P \in \mathcal{M}_{(n+1)\times(n+1)}$ is a permutation matrix, arising from a partial pivoting for instance. From this decomposition we can write

$$A = (U^t, 0)L^tP \tag{2.38}$$

and defining $y = P^t(L^t)^{-1}e_{n+1}$, for $e_{n+1} = (0, \ldots, 0, 1)^t$, then $y \neq 0$ and $Ay = 0$. From definition 2.2 $t(A) = \pm y/\|y\|$. The vector y is obtained by one backsolving and a permutation of its coordinates. The sign of y has to be set such that $\det(A^t, y) > 0$. We observe that

$$(A^t, y) = \left(P^tL\begin{pmatrix} U \\ 0^t \end{pmatrix}, P^t(L^t)^{-1}e_{n+1}\right) = P^tL\left(\begin{pmatrix} U \\ 0^t \end{pmatrix}, L^{-1}(L^t)^{-1}e_{n+1}.\right)$$

Since for $x \neq 0$ one has $x^tL^tLx = \|Lx\|^2 > 0$, then L^tL is positive definite and so is $L^{-1}(L^t)^{-1}$ $= (L^tL)^{-1}$. Hence the last entry in $L^{-1}(L^t)^{-1}e_{n+1}$ is positive and

$$\operatorname{sign}(\det(A^t, y)) = \operatorname{sign}(\det(P)\det(L)\det(U)).$$

We can easily compute the right hand side from the LU decomposition of A and from here $t(A)$. By (2.38) the matrix

$$B = P^t (L^t)^{-1} \begin{pmatrix} (U^t)^{-1} \\ 0^t \end{pmatrix}$$

is a right inverse of A and from lemma 2.5 c) it turns out that $A^+ = (I - t(A)t(A)^t)B$. Finally the obtaining of $x = A^+ b$ consists of solving two triangular systems, a permutation and a scalar product together with a sum for the orthogonal projection with $(I - t(A)t(A)^t)$.

Newton's method as a corrector. Given a predicted point $\tilde{\gamma}$ close to the curve $H(\gamma) = 0$ as in (2.37), we want to approach it finding the nearest point to $\tilde{\gamma}$ on the curve, i.e. the solution of

$$\min_{\gamma} \{ \|\gamma - \tilde{\gamma}\| \mid H(\gamma) = 0 \}.$$

A necessary condition to solve this problem is achieved by means of the method of Lagrange multipliers. Thus we want to find γ such that

$$H(\gamma) = 0, \qquad \gamma - \tilde{\gamma} = DH(\gamma)^t \lambda,$$

for some vector of multipliers $\lambda \in \mathbb{R}^{n+1}$. Because range$(A^t) = t(A)^{\perp}$, the last condition may rewritten as $t(DH(\gamma))^t(\gamma - \tilde{\gamma}) = 0$. Now, by means of Taylor's expansions of this equation and $H(\gamma) = 0$ around $\tilde{\gamma}$ up to order one, we immediately have

$$H(\tilde{\gamma}) + DH(\tilde{\gamma})(\gamma - \tilde{\gamma}) = 0, \qquad t(DH(\tilde{\gamma}))^t(\gamma - \tilde{\gamma}) = 0.$$

These equations correspond to condition a) of lemma 2.4 and hence they are equivalent to b) of the same lemma

$$\gamma = \tilde{\gamma} - DH(\tilde{\gamma})^+ H(\tilde{\gamma}), \tag{2.39}$$

which represents a step of Newton's method, modified for the case of $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$. As in the classical Newton's method, (2.39) provides local quadratic convergence (see Allgower & Georg (1990) p. 22).

Step length adaptation. A predictor method like (2.37) can give good approximations $\tilde{\gamma}(h)$ of the curve $H(\gamma) = 0$, which next will be refined by Newton's method (2.39). For the selection of the step length h is desirable to choose it as big as possible, meanwhile the corrector step achieves convergence in a a reasonable number of iterations. To this end we have adopted two strategies: by improving the precision of the predictor step, thus allowing a bigger step size h, and by adapting h according to the number of iterations of the corrector step.

Let us suppose that we have already computed points γ_0, γ_1, ..., $\gamma_k \in H^{-1}(0)$. We use a local pseudo-arclength parameterization of those points with parameter $\zeta = t^t(\gamma_i - \gamma_k)$ for $t \approx t(DH(\gamma_k))$, $\|t\| = 1$ and $i = 0, 1, \ldots, k$, in such a way that the curve $\gamma(\zeta)$ satisfies $\gamma(\zeta_i) = \gamma_i$. Next we construct the interpolating polynomial $P_{k,q}$ of degree q of those points by

$$\begin{aligned} P_{k,q}(h) = \gamma[\zeta_k] + \gamma[\zeta_k, \zeta_{k-1}](h - \zeta_k) + \gamma[\zeta_k, \zeta_{k-1}, \zeta_{k-2}](h - \zeta_k)(h - \zeta_{k-1}) \\ + \ldots + \gamma[\zeta_k, \ldots, \zeta_{k-q}](h - \zeta_k) \ldots (h - \zeta_{k-q+1}), \end{aligned} \tag{2.40}$$

where the coefficients are computed recursively by divided differences

$$\gamma[\zeta_i] = \gamma_i, \qquad \gamma[\zeta_i, \ldots, \zeta_j] = \frac{\gamma[\zeta_i, \ldots, \zeta_{j+1}] - \gamma[\zeta_{i-1}, \ldots, \zeta_j]}{\zeta_i - \zeta_j}, \quad i > j.$$

We employ the a priori estimates

$$\|P_{k,q+1}(h) - P_{k,q}(h)\| \approx \text{distance}(P_{k,q}(h), H^{-1}(0)).$$

If ε_{tol} represents the maximum allowed distance from $P_{k,q}(h)$ to $H^{-1}(0)$, according to the previous estimate, for given q we select h such that the error satisfies

$$e(h,k,q) \stackrel{\text{def}}{=} \|P_{k,q+1}(h) - P_{k,q}(h)\| = \|\gamma[\zeta_k, \ldots, \zeta_{k-q-1}]\|(h-\zeta_k)\ldots(h-\zeta_{k-q}) = \varepsilon_{\text{tol}}.$$

The error term $e(h,k,q)$ is a polynomial of degree $q+1$. In order to solve $e(h,k,q) = \varepsilon_{\text{tol}}$ for $h > 0$, we apply the secant method using as starting values $h = 0$ and $h^{q+1} = \varepsilon_{\text{tol}}/\|\gamma[\zeta_k, \ldots, \zeta_{k-q-1}]\|$. If $\bar{h}$ represents the previous step length, in order to increase the stability of the predictor, the selection of q is made as the lowest possible value, which together with h are obtained from the algorithm:

1) Set $q = 1$.
2) If $e(2\bar{h}, k, q) \leqslant \varepsilon_{\text{tol}}$ then set $h_{\text{new}} = h$ and $q_{\text{new}} = q$.
3) If condition 2) is not fulfilled we solve $e(h,k,q) = \varepsilon_{\text{tol}}$ for h, and save $h_{k,q} = h$.
4) Repeat steps 2), 3) increasing q until $h_{k,q} \geqslant h_{k,q+1}$. Then set $h_{\text{new}} = h_{k,q}$ and $q_{\text{new}} = q$.

In summary, this algorithm selects the highest new step length h_{new}, at most double of the previous one $\bar{h}$, and the lowest degree q such that $e(h_{\text{new}}, k, q) \leqslant \varepsilon_{\text{tol}}$. If the predictor-corrector procedure failed we repeated it with initial step length $\bar{h}/2$.

The value of ε_{tol} is actualized once the corrector step has converged, as a function of the number of iterations needed. Using the preceding algorithm, from $\gamma_0, \gamma_1, \ldots, \gamma_k \in H^{-1}(0)$, we obtain a predictor approximation $\sigma_0(h)$ and we call $\sigma_{i+1}(h) = C(\sigma_i(h))$ to the recurrence which carries out the corrector steps. We further suppose that, for $h > 0$ small enough, the limit $\sigma_\infty(h) = \lim_{i\to\infty} \sigma_i(h)$ exists. For certain constant $r > 0$ (independent of h) we suppose that errors $\varepsilon_i(h) = r\|\sigma_\infty(h) - \sigma_i(h)\|$ satisfy the inequalities $\varepsilon_{i+1}(h) \leqslant \psi(\varepsilon_i(h))$, where $\psi : \mathbb{R} \to \mathbb{R}$ is a known monotone function such that $\psi(0) = 0$. The more realistic function ψ for the corrector process C, the more reliable step length adaptation will be.

Let us suppose now that we want to choose $\tilde{h}$ such that the corrector step finishes in $\tilde{p}$ iterations. If for a given step length h we have needed p iterations to attain convergence, then

$$\omega(h) \stackrel{\text{def}}{=} \frac{\|\sigma_p(h) - \sigma_{p-1}(h)\|}{\|\sigma_p(h) - \sigma_0(h)\|} \approx \frac{\|\sigma_\infty(h) - \sigma_{p-1}(h)\|}{\|\sigma_\infty(h) - \sigma_0(h)\|} = \frac{\varepsilon_{p-1}(h)}{\varepsilon_0(h)} \leqslant \frac{\psi^{p-1}(\varepsilon_0(h))}{\varepsilon_0(h)},$$

from which we may estimate $\varepsilon_0(h)$ as the solution ε of $\omega(h) = \psi^{p-1}(\varepsilon)/\varepsilon$. For the new step length $\tilde{h}$ and the desired number of corrector iterations $\tilde{p}$, from the definition of ψ, we have $\varepsilon_{\tilde{p}}(\tilde{h}) \leqslant \psi^{\tilde{p}}(\varepsilon_{\tilde{p}}(\tilde{h}))$, so by imposing $\tilde{\varepsilon}$ the solution of $\psi^{\tilde{p}}(\tilde{\varepsilon}) = \psi^p(\varepsilon_0(h))$, we find an estimation of $\varepsilon_0(\tilde{h})$. $\varepsilon_0(h)$ represents the observed distance to the curve from the corrector value, meanwhile $\varepsilon_0(\tilde{h})$ corresponds to the desired distance to the curve. From here we set the new value of ε_{tol} as $\varepsilon_{\text{tol}}\varepsilon_0(\tilde{h})/\varepsilon_0(h)$.

For the particular case when the corrector process yields linear convergence, the error model is given by $\varepsilon_{i+1}(h) \leqslant \lambda\varepsilon_i(h)$ for $\lambda \in (0,1)$. Now the equation for $\varepsilon_0(h)$ is written as $\omega \approx \lambda^{p-1}\varepsilon_0(h)/\varepsilon_0(h)$ which implies $\lambda \approx \omega^{1/p-1}$. On the other hand for $\varepsilon_0(\tilde{h})$ we have $\lambda^{\tilde{p}}\varepsilon_0(\tilde{h}) = \lambda^p\varepsilon_0(h)$. Therefore it turns out

$$\frac{\varepsilon_0(\tilde{h})}{\varepsilon_0(h)} \approx \omega^{\frac{p-\tilde{p}}{p-1}},$$

as the factor to be multiplied by $\varepsilon_{\mathrm{tol}}$ to obtain the tolerance for the new corrector step.

Computation of the Jacobian matrix. Due to usual difficulties in the obtaining of the analytical Jacobian matrix $DH(\gamma)$ in (2.39), we resort to its approximation by means of finite differences. In this way for each column of $DH(\gamma)$ we select $h \in \mathbb{R}$ and estimate

$$D_jH(\gamma) \approx \frac{H(\gamma + he_j) - H(\gamma)}{h}, \tag{2.41}$$

where e_j is the vector of the canonical basis with 1 in the j^{th} position, for $j = 1, \ldots, n+1$. This is a quite economical formula in what we only have to evaluate $n+1$ additional times H to approximate the whole Jacobian $DH(\gamma)$. The proper choice of h is the main concern in finite difference approximations of derivatives. If h is too large, then (2.41) can furnish a bad approximation. Conversely, for small h cancellations in the numerator of (2.41) can be produced because $H(\gamma + he_j) \approx H(\gamma)$. A general compromise, that seems to work the best, is choosing h such that $H(\gamma + he_j)$ and $H(\gamma)$ have the first $d/2$ digits in common, if d represents the relative precision with which H is evaluated.

An effective value of h is obtained by minimizing the sum of roundoff error e_r and truncation error e_t in (2.41). If ϵ_H represents the relative precision with which H is evaluated then (2.41) leads to a roundoff error $e_r \approx \epsilon_H\|H(\gamma)/h\|$ and to a truncation error (from Taylor's formula) $e_t \approx \|hD_j^2H(\gamma)\|$. The minimum of $e_r + e_t$ is attained at $h \approx \sqrt{\epsilon_H}\gamma_c$ for $\gamma_c^2 = \|H\|/\|D_j^2H\|$. A usual approach is assuming that $\gamma_c = \gamma$.

If we approximate $D_jh(\gamma)$ by means of the central difference formula

$$D_jH(\gamma) \approx D(h) = \frac{H(\gamma + he_j) - H(\gamma - he_j)}{2h}, \tag{2.42}$$

then, as before, $e_r \approx \epsilon_f\|H(\gamma)/h\|$ and $e_t \approx \|h^2D_j^3H(\gamma)\|$. The optimal value is $h \approx (\epsilon_H)^{1/3}\gamma_c$ for $\gamma_c^3 = \|H\|/\|D_j^3H\|$. We usually take $\gamma_c = \gamma$. The use of (2.42) implies $2(n+1)$ evaluations of H to estimate DH, even though in this case e_t is better than for (2.41).

If we need even more precision, we can extrapolate (2.42) as in (2.29), whose successive application provides the recurrence

$$T_{l,0} = D(\lambda^l h), \qquad T_{l,m} = T_{l,m-1} + \frac{1}{\lambda^{-2m} - 1}\left(T_{l,m-1} - T_{l-1,m-1}\right),$$

for $0 < \lambda < 1$, $l = 0, 1, 2, \ldots$ and $m = 1, 2, \ldots, l$. We keep generating $T_{l,m}$ while $\|T_{l,m} - T_{l-1,m-1}\| = \|T_{l,m-1} - T_{l-1,m-1}\|/(1 - \lambda^{2m})$ is decreasing or small enough.

It is also worth to mention that the necessary evaluations of H in (2.41) and (2.42) are obtained using a Beowulf cluster (a set of computers working in parallel), since for each column $j = 1, \ldots, n+1$ the evaluation of $H(\gamma + he_j)$ is independent of each other.

Broyden's "good" update formula. In the corrector steps (2.39), even the numerical evaluation of DH previously described could be very expensive and, for this reason, we complement it with another approach much faster, when the cost per evaluation of H is high. We consider first a function $F : \mathbb{R}^n \to \mathbb{R}^n$ and then we extend the ideas for $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$. By means of Taylor's formula neglecting terms of order two and higher we have $A_k s_k = y_k$ for $A_k = DF(x_k)$, $s_k = x_{k+1} - x_k$ and $y_k = F(x_{k+1}) - F(x_k)$. In the same way, up to order one $A_{k+1} s_k = y_k$ for $A_{k+1} = DF(x_{k+1})$. This idea suggest that if $A_k \approx DF(x_k)$ satisfies $A_k s_k = y_k$, then we require that $A_{k+1} \approx DF(x_{k+1})$ solves the problem

$$\min_A \{\|A - A_k\|_{\mathrm{F}} \mid As_k = y_k\}, \qquad \|A\|_{\mathrm{F}}^2 = \sum_{i,j=1}^{n} a_{ij}^2.$$

By a straightforward calculation using orthogonal projections, the solution of the minimization problem is given by

$$A_{k+1} = A_k + \frac{y_k - A_k s_k}{\|s_k\|^2} s_k^t, \tag{2.43}$$

which is referred to as Broyden's "good" update formula or Broyden's rank one update. It is easy to proof (see Stoer & Bulirsch (1983) p. 267) that

$$\|A_{k+1} - DF(x_k)\|_2 \leqslant \|A_k - DF(x_k)\|_2, \qquad \|A\|_2 = \sup_{\|x\|_2=1} \|Ax\|_2,$$

where $\|\cdot\|_2$ stands for the Euclidean norm. In spite of the superlinear convergence of Broyden's method proclaimed in the next theorem (for a proof see Allgower & Georg (1990) p. 64), it need not necessary hold that $\|A_k - DF(x_k)\| \to 0$ as $k \to \infty$.

Theorem 2.6. *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a smooth map and $\bar{x} \in \mathbb{R}^n$ be a regular zero point of F. Suppose $x_{k+1} = x_k - A_k^{-1} F(x_k)$ is the Newton-type method where A_k is updated according to Broyden's formula* (2.43)*. If x_0 is sufficiently near $\bar{x}$, and if A_0 is sufficiently near $DF(x_0)$, then the sequence $\{x_k\}$ converges superlinearly to $\bar{x}$ i.e.*

$$\lim_{k\to\infty} \frac{\|x_{k+1} - \bar{x}\|}{\|x_k - \bar{x}\|} = 0.$$

The above ideas can be applied to traversing of the curve $H^{-1}(0)$. As before we consider two approximate zero points γ_k, γ_{k+1} and put $s_k = \gamma_{k+1} - \gamma_k$, $y_k = H(\gamma_{k+1}) - H(\gamma_k)$. If A_k satisfies the secant equation $A_k s_k = y_k$, we update A_{k+1} by means of (2.43). The following properties are demonstrated in Allgower & Georg (1990) p. 70.

Theorem 2.7. *Suppose $a_k \in \mathbb{R}^n$, $b_k \in \mathbb{R}^{n+1}$ and $A_k \in \mathcal{R}_n = \{B \in \mathcal{M}_{n\times(n+1)} \mid \operatorname{rank}(B) = n\}$. Define $A_{k+1} = A_k + a_k b_k^t$, $D_k = 1 + b_k^t A_k^+ a_k$, $t_k = t(A_k)$ and assume $D_k \neq 0$. Then:*

1) $\operatorname{rank} A_{k+1} = n$.
2) $t_{k+1} = r_k(t_k - b_k^t t_k A_k^+ a_k / D_k)$ *for some* $r_k \in \mathbb{R}$ *with* $|r_k| \in (0, 1]$.
3) $A_{k+1}^+ = (I - t_{k+1} t_{k+1}^t)(I - A_k^+ a_k b_k^t / D_k) A_k^+$.
4) $\det \bar{A}_{k+1} = D_k \det \bar{A}_k / r_k$, *where $\bar{A}$ means matrix A completed with $t(A)^t$ in the last row.*

In order to apply (2.43), following the notation of the theorem, we put $a_k = (y_k - A_k s_k)/\|s_k\|$ and $b_k = s_k/\|s_k\|$. We have to distinguish between predictor and corrector steps. For the first case $\gamma_{k+1} = P_{j,q}(h)$ for $P_{j,q}$ as defined in (2.40) and $j, q \in \mathbb{N}$ with $q \leqslant j$. For the corrector step as in (2.39) we have $\gamma_{k+1} = \gamma_k - A_k^+ H(\gamma_k)$, which yields $a_k = H(\gamma_{k+1})/\|A_k^+ H(\gamma_k)\|$, $b_k = -A_k^+ H(\gamma_k)/\|A_k^+ H(\gamma_k)\|$ and $A_{k+1} = A_k - H(\gamma_{k+1})(A_k^+ H(\gamma_k))^t/\|A_k^+ H(\gamma_k)\|^2$. Likewise from $b_k^t t_k = 0$ it results $t_{k+1} = \pm t_k$. Furthermore $\|A_k^+ a_k\| = \|A_k^+ H(\gamma_{k+1})\|/\|A_k^+ H(\gamma_k)\|$ gives a reasonable measure for the contraction rate of the corrector step. It can be large because either the predictor point was too far from $H^{-1}(0)$ or $DH(\gamma_k)$ was poorly approximated by A_k. From here we have an estimation of the quality of the approximation of the Jacobian which should be monitored at each update.

Detection of simple bifurcation points. We are going to describe how to detect simple bifurcation points along the curve $\gamma(s) \in H^{-1}(0)$. At these points $\operatorname{rank}(DH) < n$.

Definition 2.8. *Let $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$ be sufficiently smooth. Suppose that $\gamma(s) \in \mathbb{R}^{n+1}$ is a smooth curve for $s \in J$ the parameter arclength, and $J \ni 0$ some open interval, such that $H(\gamma(s)) = 0$ for $s \in J$. The point $\gamma(0)$ is called a bifurcation point of $H = 0$ if there exists $\varepsilon > 0$ such that every neighborhood of $\gamma(0)$ contains zero-points of H which are not on $\gamma(-\varepsilon, \varepsilon)$.*

Definition 2.9. *Let $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$ be sufficiently smooth. A point $\bar{\gamma} \in \mathbb{R}^{n+1}$ is called a simple bifurcation point of $H = 0$ if the following conditions hold: a) $H(\bar{\gamma}) = 0$, b) $\dim \ker DH(\bar{\gamma}) = 2$, and c) $e^t DH^2(\bar{\gamma})|_{(\ker DH(\bar{\gamma}))^2}$ has one positive and one negative eigenvalue, where e spans $\ker DH(\bar{\gamma})$.*

The proof of the following theorems can be found in Allgower & Georg (1990) p. 79.

Theorem 2.10. *Let $\bar{\gamma} \in \mathbb{R}^{n+1}$ be a simple bifurcation point of $H = 0$. Then there exists two smooth curves $\gamma_1(s)$, $\gamma_2(s) \in \mathbb{R}^{n+1}$ parameterized with respect to arclength s, defined for $s \in (-\varepsilon, \varepsilon)$ and ε sufficiently small, such that the following holds: a) $H(\gamma_i(s)) = 0$, $i = 1, 2$, $s \in (-\varepsilon, \varepsilon)$, b) $\gamma_i(0) = \bar{\gamma}$, $i = 1, 2$, c) $\dot{\gamma}_1(0)$ and $\dot{\gamma}_2(0)$ are linearly independent, and d) $\bar{\gamma}$ is not in the closure of $H^{-1}(0) \backslash (\operatorname{range}(\gamma_1) \cup \operatorname{range}(\gamma_2))$.*

Theorem 2.11. *Let $\bar{\gamma} \in \mathbb{R}^{n+1}$ be a simple bifurcation point of $H = 0$. Then the determinant of the augmented Jacobian*

$$\det \begin{pmatrix} DH(\gamma_i(s)) \\ \dot{\gamma}_i(s)^t \end{pmatrix},$$

changes sign at $s = 0$ for $i = 1, 2$.

This theorem furnishes a criteria to detect simple bifurcation points of $H^{-1}(0)$. The determinant of the augmented Jacobian is easily obtained from decomposition (2.38).

Searching of special points along the curve. Let us consider a smooth function $f : \mathbb{R}^{n+1} \to \mathbb{R}$ defined on points $\gamma \in \mathbb{R}^{n+1}$ such that $H(\gamma) = 0$ for $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$ a given smooth mapping. We suppose that the curve $H^{-1}(0)$ is parameterized with the arclength s. First we want to find $\gamma \in H^{-1}(0)$ such that $f(\gamma) = 0$. To that end we define $g(s) = f(\gamma(s))$ and search for solutions of $g(s) = 0$. Given $\gamma(s_1), \gamma(s_2) \in H^{-1}(0)$, we put $g_i = g(s_i)$ for $i = 1, 2$ and applying the secant method, for the previous steplength $h = s_2 - s_1$, we impose

$$g_2 + \frac{g_2 - g_1}{h}(s - s_2) = 0 \quad \Longrightarrow \quad h_{\text{new}} = s - s_2 = \frac{g_2}{g_1 - g_2} h, \tag{2.44}$$

where h_{new} is the new steplength to obtain $\gamma(s_2 + h_{\text{new}})$ as the approximated solution of $g(s) = 0$.

As a second problem we want to find a local minimum of $h(s) = f(\gamma(s))$ and thus we look for solutions of $h'(s) = f'(\gamma(s))\gamma'(s) = 0$. In particular we are interested in the case when $f(\gamma) = \gamma_p$ where $\gamma = (\gamma_1, \ldots, \gamma_{n+1})$ and $p \in \{1, \ldots, n+1\}$. Defining $g(s) = \gamma_p'(s)$, it corresponds to the p^{th} coordinate of the tangent vector at $\gamma(s)$. We approximate the solution of $g(s) = 0$ by applying again the secant method (2.44).

2.9 Minimization without derivatives

Following we describe the approach employed to obtain the minimum Re for which exists a rotating wave and likewise, given an approximate fixed point of a Poincaré map as defined in §4.1, how to obtain the best value of c that makes it a modulated wave. The procedure is based on first bracketing the minimum and then a golden section search method combined with inverse parabolic interpolation (for more details see Press, Teukolsky, Vetterling & Flannery 1992, chapter 10).

We consider a function $f : \mathbb{R} \to \mathbb{R}$ and we want to find a triplet (a, b, c) such that $f(b) < f(a)$ and $f(b) < f(c)$: this is called the bracketing of a minimum. If $f(b) < f(a)$ we proceed in the direction $b - a$ taking larger steps meanwhile f is decreasing. Given c such that $f(c) < f(b) < f(a)$ we extrapolate the minimum by the interpolating polynomial of f at the points a, b and c. We combine a constant increase in the descendent direction with parabolic extrapolation, until a minimum is bracketed.

If the triplet (a, b, c) with $a < b < c$ bracket the minimum m of f, the best choice for a better approximation of m is selected taking $x = c - (b - a)$ as is checked by a simple equation. If we always follows this policy, we find that the constant ratio w at which x must be placed satisfies the quadratic equation $w^2 - 3w + 1 = 0$, whose solution $w \approx 0.38197$ is closely related to the so called golden mean. If the starting triplet (a, b, c) has not the golden ratio, the procedure of choosing successive points at the golden mean point will quickly converge to the golden ratio. The reduction rate of the uncertainty interval is 0.38197 at each iteration. We accelerate the convergence by taking advantage of parabolic extrapolations as in Brent's method described in Press *et al.* (1992).