

Stability Properties of POD–Galerkin Approximations for the Compressible Navier–Stokes Equations*

Angelo Iollo, Stéphane Lanteri, and Jean-Antoine Désidéri

Projet SINUS, INRIA Sophia Antipolis, France
{Angelo.Iollo, Stephane.Lanteri, Jean-Antoine.Desideri}@inria.fr

Communicated by M.Y. Hussaini

Received 21 April 1999 and accepted 18 November 1999

Abstract. Fluid flows are very often governed by the dynamics of a small number of coherent structures, i.e., fluid features which keep their individuality during the evolution of the flow. The purpose of this paper is to study a low order simulation of the Navier–Stokes equations on the basis of the evolution of such coherent structures. One way to extract some basis functions which can be interpreted as coherent structures from flow simulations is by Proper Orthogonal Decomposition (POD). Then, by means of a Galerkin projection, it is possible to find the system of ODEs which approximates the problem in the finite-dimensional space spanned by the POD basis functions. It is found that low order modeling of relatively complex flow simulations, such as laminar vortex shedding from an airfoil at incidence and turbulent vortex shedding from a square cylinder, provides good qualitative results compared with reference computations. In this respect, it is shown that the accuracy of numerical schemes based on simple Galerkin projection is insufficient and numerical stabilization is needed. To conclude, we approach the issue of the optimal selection of the norm, namely the H^1 norm, used in POD for the compressible Navier–Stokes equations by several numerical tests.

1. Introduction

The natural ambition of the aerodynamic designer is to influence the structure of the flow (for example, by local injection of fluid) with the aim of increasing or optimizing performance. For this reason automatic shape design and active flow control are objects of intense theoretical and practical research. One central problem in applications is the prohibitive amount of computational resources needed by repeated application of a flow solver, in particular in the case of three-dimensional, unsteady (possibly turbulent) flows. A promising technique to circumvent this difficulty is the adoption of low order models as governing equations. Such models should provide a qualitative description of the main features of the flow, for example, in terms of lift and drag versus time, while permitting a very economical numerical solution.

One way to devise a low order model is to use Proper Orthogonal Decomposition (POD), by which it is possible to extract from a database of direct simulations a certain number of basis functions onto which the Navier–Stokes equations are projected. The Navier–Stokes equations are thus reduced to a finite-dimensional system of nonlinear ODEs. If the dimension of this system is reasonably small, the solution can be found with very limited computational effort.

The POD was introduced in fluid mechanics by Lumley [8] in the study of turbulent flows (for a review see [4]). In that setting, the basis functions obtained by POD were recognized to be coherent structures,

* The first author gratefully acknowledges the support of the Marie Curie Fellowship grant.

i.e., spatial features which repeatedly appear in space and time. In those works, the basic assumption was that a limited number of coherent structures capture most of the dynamics of the system, relying on the fact that turbulence might be the manifestation in the physical space of a strange attractor of limited dimension in phase space. We retain in the following the same assumption, i.e., that a limited number of coherent structures (about 50 in our cases) suffices to describe the flow. In fact, this is verified *a posteriori*.

It may be objected that such an approach is of limited applicability since the basis functions obtained for a given configuration or parameter regime are not suitable for other simulations. Typically, this is the case when the control is acting on the flow, so that new coherent structures, not present in the uncontrolled flow, appear. Although this is true in general, there are many examples in which the coherent structures do not qualitatively change for wide ranges of the flow parameters. In addition, there are techniques to enlarge the computational database in order to include the effect of controls. For example, it is proposed in [13] to simulate the flow under the action of a random control to generate a sufficiently rich database, which could also encompass the coherent structures of the specific control subsequently used.

In this study we contribute in two ways to the construction of a robust scheme for the integration of the system of ODEs resulting from the projection of the compressible Navier–Stokes equations. Firstly, on the basis of what was done in [7] for the Euler equations, we show by numerical evidence that when a low order model is constructed using POD to describe the main flow features, it is not necessary to model the unresolved modes in some alternate way to achieve numerical stability. Aubry *et al.* [4] presented a model for the flat plate turbulent boundary layer based on a Galerkin projection of the Navier–Stokes equations onto POD eigenfunctions resulting from experiments. The unresolved modes have been accounted for by a model for the small scales which ensures the proper amount of dissipation (and stabilization), so that the system of ODEs obtained is stable. The energy budget of such a small scales model correctly reproduces the dissipation due to neglected modes, as found by Berkooz *et al.* [5]. Also, in [3] a paragraph is dedicated to the dissipation properties of truncated POD models to the Kuramoto–Sivashinsky equation. The main observation is again that if the POD modes considered are not enough, the solution may blow up in finite time.

However, in our perspective, POD can only enlarge the stability domain of the numerical scheme in the complex plane. This is found by classical numerical analysis tools for a one-dimensional advection–diffusion equation and verified by some numerical experiments for an unsteady laminar flow, in which a textbook four-stage Runge–Kutta time integration scheme advances with a CFL number of 75! We conclude that numerical stabilization of POD–Galerkin approximation of the Navier–Stokes equations can be attained by judicious use of numerical dissipation as an alternative to modeling the unresolved modes as was proposed in [2].

The second contribution of this work is the discussion concerning the canonical L^2 norm in the definition of POD. For example, to retain the optimality of the POD basis functions, we show that in the simple case of the linear advection–diffusion equation with Dirichlet conditions, it is preferable to include a weighting function in the definition of the scalar product. In the case of the turbulent Navier–Stokes equations, a numerical test involving POD basis functions obtained using the H^1 formulation, showed that the resulting low order model is more dissipative and hence less subject to numerical instability than the regular L^2 formulation. Norms other than L^2 were already proposed in [1] basically to give an appropriate functional setting for noncompact kernels relative to biorthogonal expansions. However, the context and the purposes for which they were introduced are different from those characterizing our work.

We anticipate that greater sophistication in the device employed for numerical stability will ultimately contribute to applying optimal flow control techniques to the compressible Navier–Stokes equations as successfully as in the incompressible case, for which more results have yet to be established in the literature.

The article is organized as follows. In Section 2 POD basis functions are determined in the setting of the Hilbert–Schmidt theory of kernels. In Section 3 we analyze POD approximation of the one-dimensional advection–diffusion with different boundary conditions. We show that POD can be interpreted as a filter and we compare the constant coefficients system of ODEs with POD filtering. In Section 4 we present the numerical experimentation setting and the results obtained with the POD filter for the case of the flow past an NACA0012 at 20° of incidence, $Mach = 0.2$, $Re = 2100$ (laminar flow), and for the flow about a square cylinder, $Mach = 0.1$, $Re = 22000$ (turbulent flow). For the same test cases we present the results obtained with the H^1 POD filter. To conclude, the results obtained with the constant coefficients system of ODEs are compared with those obtained by POD filtering.

2. POD of the Compressible Navier–Stokes Equations

The method of “snapshots” proposed by Sirovich [12] for the incompressible Navier–Stokes equations is adapted here to the compressible Navier–Stokes equations. This method, briefly outlined below for completeness, allows for the construction of the basis functions starting from a limited number of “snapshots” of the flow at successive times.

Consider the nondimensional Navier–Stokes equations in two dimensions written in conservative form with $W = \{\rho, \rho u, \rho v, \rho e\}$:

$$W_t + F_x + G_y = \frac{1}{Re} (R_x + S_y); \quad (1)$$

$$F = \{\rho u, \rho u^2 + p, \rho uv, u(\rho e + p)\} \quad \text{and} \quad G = \{\rho v, \rho uv, \rho v^2 + p, v(\rho e + p)\}$$

denote the convective fluxes while

$$R = \left\{ 0, \frac{4}{3} u_x - \frac{2}{3} v_y, v_x + u_y, u \left(\frac{4}{3} u_x - \frac{2}{3} v_y \right) + v (v_y + u_x) + \frac{\gamma}{\gamma - 1} \frac{T_x}{Pr} \right\}$$

and

$$S = \left\{ 0, v_x + u_y, \frac{4}{3} v_y - \frac{2}{3} u_x, v \left(\frac{4}{3} v_y - \frac{2}{3} u_x \right) + u (v_y + u_x) + \frac{\gamma}{\gamma - 1} \frac{T_y}{Pr} \right\}$$

denote the diffusive fluxes. In the above expressions ρ is the density, u and v are the cartesian components of the flow speed, e is the total energy per unit mass, and T is the temperature. The pressure is given by $p = 1/2(\gamma - 1)(2\rho e - \rho u^2 - \rho v^2)$ and γ is the specific heat ratio.

Suppose that, from existing data, the solution at M different time steps t_n is known in terms of $W^{(n)} = W(x, y; t_n)$. It is required to find a function

$$\phi(x) = (\phi_1, \phi_2, \phi_3, \phi_4) \in (L^2(R^2))^4$$

which gives the best representation of the set of $W^{(n)}$ in the following sense:

$$\frac{\langle (W^{(n)}, \phi)^2 \rangle}{(\phi, \phi)} = \max_{\psi} \frac{\langle (W^{(n)}, \psi)^2 \rangle}{(\psi, \psi)}, \quad (2)$$

where (ψ, ψ) denotes the canonical L^2 inner product and the brackets $\langle \cdot \rangle$ indicate the time average

$$\langle (W^{(n)}, \psi)^2 \rangle = \frac{1}{M} \sum_{n=1}^M (W^{(n)}, \psi)^2. \quad (3)$$

In other words, one seeks a function ϕ which is most parallel (correlated) in an average sense to the given solution set $W^{(n)}$.

This problem finds its natural setting in the theory of linear integral equations (see Chapter III of [6]). Here some results relevant to our study are sketched in the frame of such a theory. For omitted proofs we refer the reader to [6].

Consider first the general quadratic integral form

$$J(\psi, \psi) = \int_{\Omega} \psi(s) K(s, t) \psi(t) dt ds, \quad (4)$$

where $K(s, t) = K(t, s)$ is a symmetric 4×4 matrix and $\psi \in (L^2(R^2))^4$. Suppose that the functional J takes on positive values only, i.e., $\forall \psi, J(\psi, \psi) > 0$; in this case $J(\psi, \psi)$ is said to be positive definite. Suppose also that $(\psi, \psi) = 1$.

If the function $\phi^{(l)}$ satisfies

$$\int_{\Omega} K(s, t) \phi^{(l)}(t) dt = \lambda_l \phi^{(l)}(s), \quad (5)$$

then clearly $J(\phi^{(l)}, \phi^{(l)}) = \lambda_l$, and it can be shown that $\phi^{(l)}(s)$ realizes a local maximum of $J(\psi, \psi)$ subject to the constraint $(\psi, \psi) = 1$.

Now, with the particular choice of the kernel

$$K(s, t) = \frac{1}{M} \sum_{n=1}^M W^{(n)}(s) \otimes W^{(n)}(t), \quad (6)$$

where \otimes is the tensorial product, we have

$$J(\psi, \psi) = \frac{1}{M} \sum_{n=1}^M \left[\int_{\Omega} \psi(s) W^{(n)}(s) \, ds \int_{\Omega} \psi(t) W^{(n)}(t) \, dt \right] = \langle (W^{(n)}, \psi)^2 \rangle. \quad (7)$$

Therefore the maximum of $J(\psi, \psi)$ with $(\psi, \psi) = 1$ solves the problem in (2). Substituting (6) into (5) we have

$$\frac{1}{M} \sum_{n=1}^M \int_{\Omega} W^{(n)}(s) \otimes W^{(n)}(t) \phi^{(l)}(t) \, dt = \lambda_l \phi^{(l)}(s). \quad (8)$$

This is an eigenvalue problem for the symmetric, degenerate, positive definite kernel in (6). In this respect, recall that every continuous symmetric kernel that does not vanish identically possesses eigenvalues and eigenfunctions. Also since the kernel in (8) is real, symmetric, and degenerate then the eigenvalues are M in number and all real. The eigenvalues are positive since $J(\psi, \psi)$ in (7) is positive definite.

The totality of eigenvalues and eigenfunctions of (8) are found sequentially. Each new eigenfunction is sought as the solution of the maximization problem for the functional J subject to the constraint of being orthogonal to all previously found eigenfunctions. Any r -fold degenerate eigenvalue is associated with r linearly independent eigenfunctions.

In conclusion, once the eigenvalue problem of (8) is solved, we are left with a set of M eigenvalues λ_n and M orthonormal eigenfunctions $\phi^{(n)}$ which give an optimal representation, in the sense made precise at the beginning of this section, of the solution known at different time steps. In the following sections we use similar arguments to determine a basis for the approximation of the compressible Navier–Stokes equations.

Sirovich [12] proposed an efficient and simple way to solve the eigenvalue problem based on the fact that the eigenfunctions can be expressed in terms of the original set of data, i.e.,

$$\phi(s) = \sum_{n=1}^M \chi_n W^{(n)}(s), \quad (9)$$

where the χ_n ($n = 1, \dots, M$) are scalar coefficients. Substituting (9) in (8) results in a linear algebra problem consisting of finding the eigenvalues and the eigenvectors of

$$C \mathcal{K} = \lambda \mathcal{K}, \quad (10)$$

where $\mathcal{K} = (\chi_1, \dots, \chi_M)$ is one of the M eigenvectors, λ is the corresponding eigenvalue, and $C = 1/M (W^{(i)}, W^{(j)})$ for $i, j = 1, \dots, M$. In the applications of the following sections we make use of the POD-functions gradients, which we compute simply by

$$\nabla \phi(s) = \sum_{n=1}^M \chi_n \nabla W^{(n)}(s). \quad (11)$$

When the solution W is sampled by means of a large number of snapshots, a certain number of eigenvalues are very likely to be close to zero, so that the contribution of the corresponding eigenfunction to the description of the flow may be considered negligible. In this case we can consider only the first M_c (normally about 50) eigenvalues for the projection of the governing equations over the corresponding eigenfunctions, in order to reduce the set of ODEs to be solved. In practical applications, we normally solve this eigenproblem by the use of NAG library routines.

Before considering applications to flow simulations, in the next section we analyze general properties of the POD approach when applied to single model problems representative of flow equations.

3. Analysis of POD Applied to Model Equations

We consider first the periodic solutions of the pure advection equation

$$u_t + u_x = 0$$

with $x \in [0, 2\pi]$. A semidiscretization over the uniform mesh $x_i = i 2\pi/N$ ($i = 0, \dots, N$) results in the following set of ODEs (“method of lines”):

$$U_t + \frac{N}{2\pi} A U = 0, \quad (12)$$

where $U = \{u(x_i)\}$ and $A = \{a_{ni}\}$ is the spatial discretization operator. In this case the POD basis functions are (unsurprisingly) the usual Fourier modes

$$\phi^k(x_i) = \frac{1}{\sqrt{N}} \exp \left[\frac{ik2\pi i}{N} \right],$$

so that substituting in (12) the expression

$$U = \sum_{k=-M}^M w^k(t) \{\phi^k(x_i)\}, \quad (13)$$

and making the l^2 (discrete) projection over the mode $\bar{\phi}^j$ yields the following reduced set of ODEs:

$$w_t^j + \sum_{k=-M}^M w^k \{\bar{\phi}^j(x_n)\} \{a_{ni}\} \{\phi^k(x_i)\} = 0, \quad j = 1, \dots, M. \quad (14)$$

The matrix $\Lambda = \{\lambda_{jk}\} = \{\{\bar{\phi}^j(x_n)\} \{a_{ni}\} \{\phi^k(x_i)\}\}$ is diagonal as the $\{\phi^j(x_i)\}$ are its eigenvectors. On the diagonal of Λ appear the eigenvalues of A , the discrete operator.

From (14), multiplying by ϕ^j and subsequently summing over j , we reconstruct the PDE in the physical space. We obtain

$$U_t + \frac{N}{2\pi} \tilde{A} A U = 0 \quad (15)$$

with $\tilde{A} = \{\tilde{a}_{ni}\} = \{\sum_{k=-M}^M \bar{\phi}^k(x_n) \phi^k(x_i)\}$. The matrix \tilde{A} , which is circulant and symmetric, has the same eigenvectors as the matrix A . Its first M eigenvalues (not to be confused with the eigenvalues of the correlation matrix which yields the POD basis functions) are 1, whereas the remainder are 0. Therefore, the first M eigenvalues of the spatial operator $\tilde{A} A$ are equal to those of A , the others are canceled out.

For example, suppose that the spatial operator is the usual upwind differentiation, and suppose that a Runge–Kutta scheme is used to integrate (15) forward in time. Since the higher frequencies are filtered out by the matrix \tilde{A} , the maximum CFL allowed is higher, see Figure 1. In this figure the eigenvalues of the upwind spatial operator are plotted in the complex plane. They lie over a circle \mathcal{C} of radius 1 tangent to the imaginary axis, the lower frequencies corresponding to the eigenvalues closer to the origin. Suppose we map \mathcal{C} onto a circle \mathcal{C}' which is tangent to the domain of stability of the time-integration scheme, e.g., the four-stage Runge–Kutta scheme, by an homothetic transform. The radius of \mathcal{C}' is equal to the maximum allowed CFL number. Clearly, the smaller the number of retained frequencies, the higher the radius of \mathcal{C}' . In the following section we show that this is experimentally verified also in the case of the unsteady Navier–Stokes equations.

This simple analysis should clarify why the POD model is not responsible for the stability degradation of the numerical scheme observed, for example, in [16]. We ascribe the numerical stability degradation of the cases that we studied to an improper treatment of the convective fluxes derivatives in the POD model.

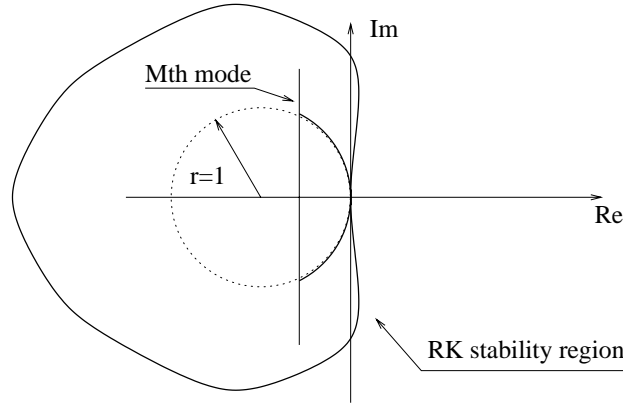


Figure 1. Plane of the eigenvalues of $\tilde{A}A$. In this case A is the upwind operator. For an explicit time integration the CFL bounds are increased.

Also it is seen that POD can be viewed as a filter possibly adapted to the flow under consideration. We use this interpretation to check if fairly complex flows can be conveniently represented by a reduced number of basis functions.

The same type of analysis can be applied with some caution to other linear spatial operators with different boundary conditions. For example, if we consider the advection–diffusion equation with homogeneous Dirichlet boundary conditions

$$u_t + c u_x - \nu u_{xx} = 0,$$

if central differencing is used

$$A = \text{Trid}(-\nu/h^2 - c/2h, 2\nu/h^2, -\nu/h^2 + c/2h),$$

where h is the mesh size. The m th eigenvector of matrix A is then given by its components

$$e^m = \{e_j\}^m = \alpha^j \sin(j\theta_m); \quad \alpha = \sqrt{\frac{-\nu/h^2 - c/2h}{-\nu/h^2 + c/2h}},$$

$\theta_m = m\pi/(N + 1)$ is the frequency parameter, N is the number of discretization points, and $j = 1, \dots, N$. These eigenvectors approximate the eigenfunctions of the continuous spatial operator with Dirichlet boundary conditions, i.e., $\exp(cx/2\nu) \sin(m\pi x)$. However, the eigenfunctions of the continuous spatial operator as well as their discrete approximations are orthogonal only if an appropriate weighted scalar product is used. In the discrete case the weight is α^{-2j} and we have, for example,

$$(e^m, e^n) = \sum_j e_j^m e_j^n \alpha^{-2j} = \delta_{mn}.$$

Similarly, if the POD basis functions are to recover the advantageous decoupling of (12), it is necessary that, in the definition of (2), the scalar products are defined accordingly using the same weighting function as the continuous operator, i.e., $\exp(-cx/4\nu)$ or its discrete counterpart. In this way the POD basis functions can be found to be the discrete approximations of the eigenfunctions of the continuous operator, so that matrices \tilde{A} and A commute the same as in the previous example.

In this linear case we see that the use of a more sophisticated euclidean norm induced by the discrete operator A , dramatically improves the approximation properties of the basis functions found by (2). In the next section we show an attempt to extend such ideas to the compressible Navier–Stokes equations.

We conclude with a nonlinear example. Consider the case of the inviscid Burgers' equation

$$u_t + u u_x = 0 \tag{16}$$

with $u = \sum_{k=1}^M w^k(t) \phi^k(x)$. Projecting onto $\phi^j(x)$ yields

$$w_t^j + \sum_{k=1}^M w^k w^j (\phi_x^k, \phi^j) = 0. \tag{17}$$

Alternatively, using POD as a filter one obtains

$$U_t + \tilde{A} A(U) U = 0. \tag{18}$$

In (17) the term (ϕ_x^k, ϕ^j) can be computed once and for all, so that the solution of this system of ODEs is very efficient. In this specific case, assuming an implicit time integration, the system is stable. Yet, if we proceed in the same way for the Euler [7] and the compressible Navier–Stokes equations [16] the resulting system of ODEs is unstable. The numerical stabilization of these ODE systems is still an open question, although in [16] several interesting experiments were performed by adding artificial diffusion.

The POD-filter approach is the dual of the constant coefficients ODE system. It is easily implemented and stable but inefficient. Nevertheless, it shows that low order modeling does not necessarily imply instability, and it provides a simple way to study the quality of the approximation which can be obtained by a low order POD–Galerkin formulation.

4. Compressible Flow Simulations

With reference to (1), we take $W = \sum_{j=1}^M w^j(t) \phi^j(x)$ where $w^j(t) \in L^2(R)$. In the case of the Euler and laminar Navier–Stokes equations, $\phi^j \in (L^2(R^2))^4$, whereas for turbulent flows modeled using the Reynolds averaged Navier–Stokes equations, we have $\phi^j \in (L^2(R^2))^6$ in order to account for κ and ε . In all of the computations we took $Pr = 0.72$ and $\gamma = \frac{7}{5}$. The code on which we base our analysis and computational results combines the following features: a mixed finite volume/finite element formulation on unstructured triangular meshes; the convective part of the Euler equations is discretized using an upwind finite volume formulation based on an approximate Riemann solver [11]. Second-order spatial accuracy is obtained through a MUSCL [15] technique which requires the construction of local solution slopes; the latter being computed using a classical finite element formulation for P1 triangular elements. Linearized implicit formulation for second-order time integration using a defect correction approach [9] is applied. For turbulent flows we resort to a κ – ε model with wall laws. For the details concerning the test cases that we will present, we refer the reader to [10] and [14]. They were selected as representatives of typical unsteady flows which may be interesting to control.

4.1. POD Filter

Based on the code we described above, we use the POD filter constructed in a way similar to that explained in the previous sections. At each time step we compute the flow variables update using the reference code, then we filter and start the process all over again as explained in Figure 2. The POD basis functions are obtained by the “snapshots method.”

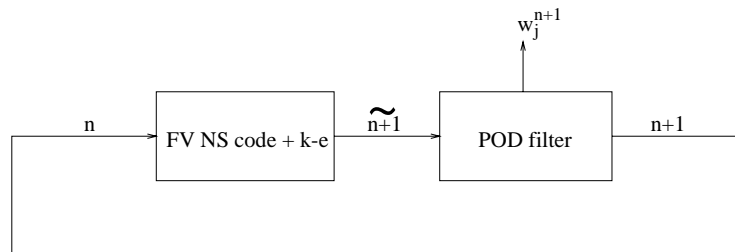


Figure 2. Time integration scheme with a POD filter.

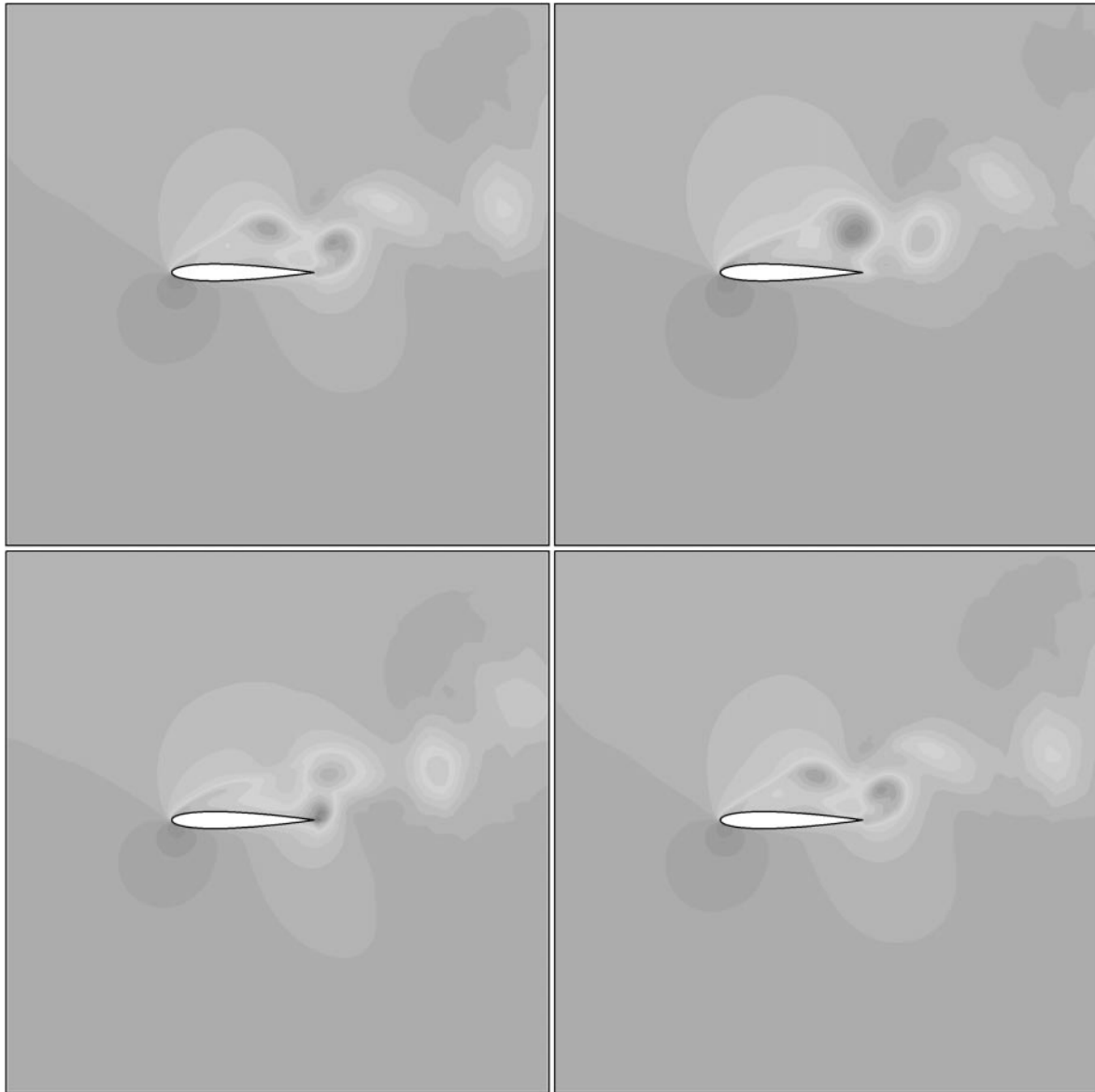


Figure 3. NACA0012: 20° of incidence, Mach = 0.2, $Re = 2100$, laminar. Example of snapshots: density.

The first case we consider is an NACA0012 at 20° of incidence immersed in a flow at Mach = 0.2 and $Re = 2100$. The flow field is discretized over a triangular mesh of 12 248 vertices. The flow over the airfoil is laminar with periodic vortex shedding from the leading edge and from the trailing edge, see Figure 3. We sample such a flow with about 100 snapshots over one period. Then we construct the basis functions and retain 30 of them, those corresponding to the highest eigenvalues. In Figure 4 we see the first component of the second to fifth basis function. Notice that the first basis function is not shown as in our case it is related to the average flow. It is seen that a phase shift appears between the second and third basis functions as well as between the fourth and fifth, whereas the length scale of the structures is constant from the first to the last one. It may be noticed that the vortex street past the airfoil is out of phase in the first couple of basis functions and in phase in the latter. Indeed, the corresponding eigenvalues also exhibit such a pairing. The length scale of the spatial structures decreases with the subsequent basis functions, not shown here. The pairing of the eigenvalues and the phase shift between coupled POD modes has been shown in [1] to be a consequence of the translation symmetry in the traveling waves characterizing the flow.

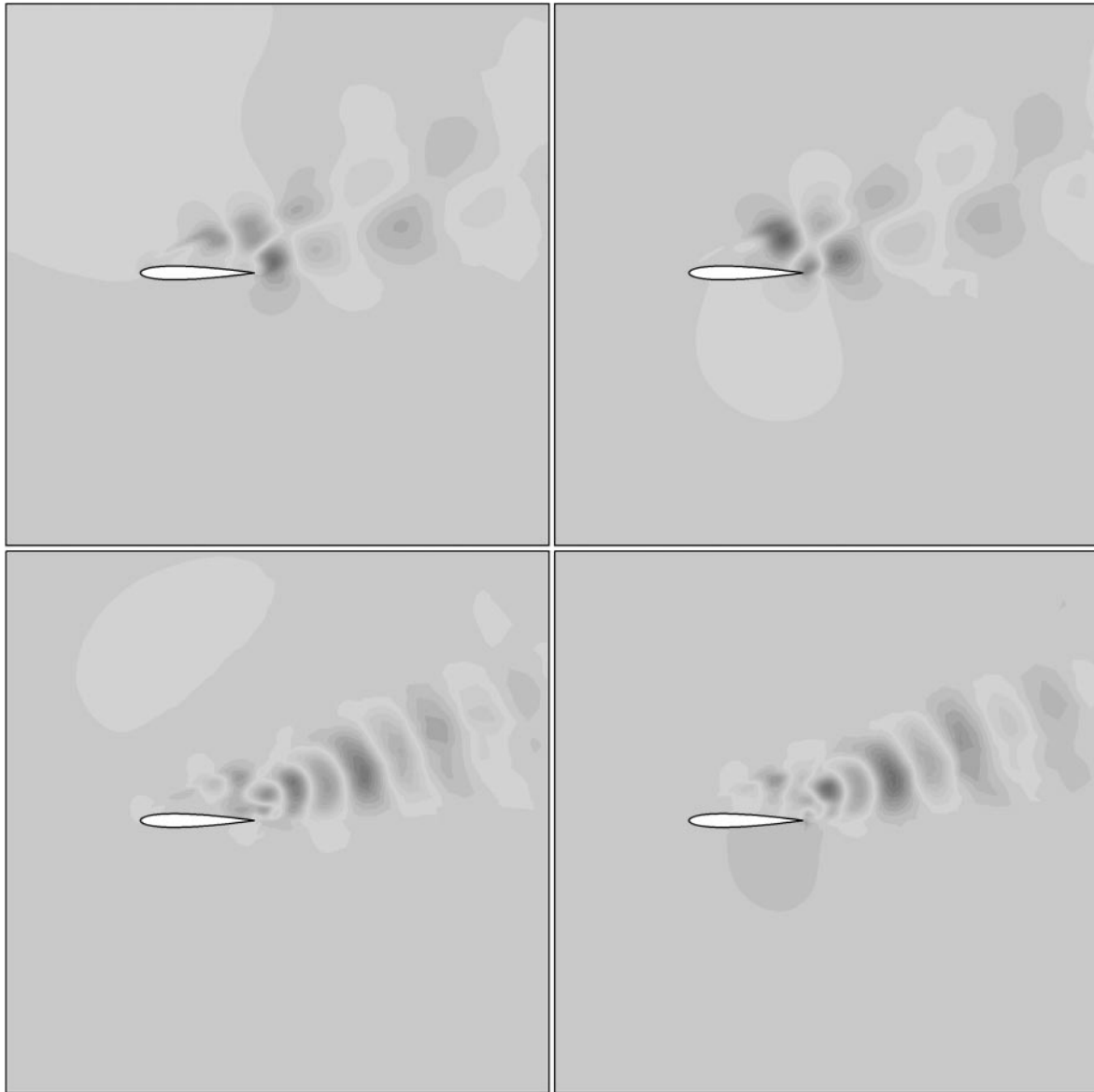


Figure 4. Example of basis functions for the laminar case: first component.

In Figure 5 we compare the results of the reference code with those obtained using the POD filter. The results are given in terms of lift versus time; there is a slight increase in the oscillation amplitude, which stays, however, bounded for all times, more remarkable as the number of basis functions used is decreasing as is seen in Figure 6. The lift plateau due to the vortex shedding from the trailing edge is in all cases except the last overestimated. The quality of the simulation is reasonable when ten basis functions or more are used.

The explicit reference code, which realizes the time integration with a classic four-stage Runge–Kutta method, operates with a maximum CFL of about 2.5, whereas in the context of the POD filter code the same Runge–Kutta routine can be stably operated with a maximum CFL of about 75, see Figure 6. It is remarkable that in accordance with the one-dimensional linear analysis the POD filter code has a stability domain of larger extent.

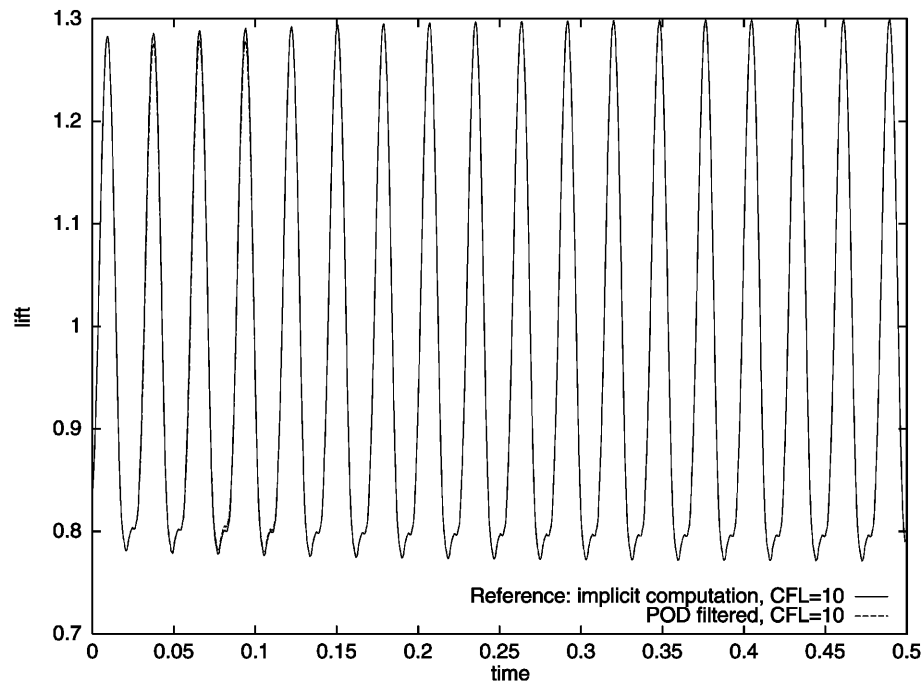


Figure 5. Results of the reference code compared with the ones obtained by POD filtering in the laminar case. The reference computation was pursued for only four periods.

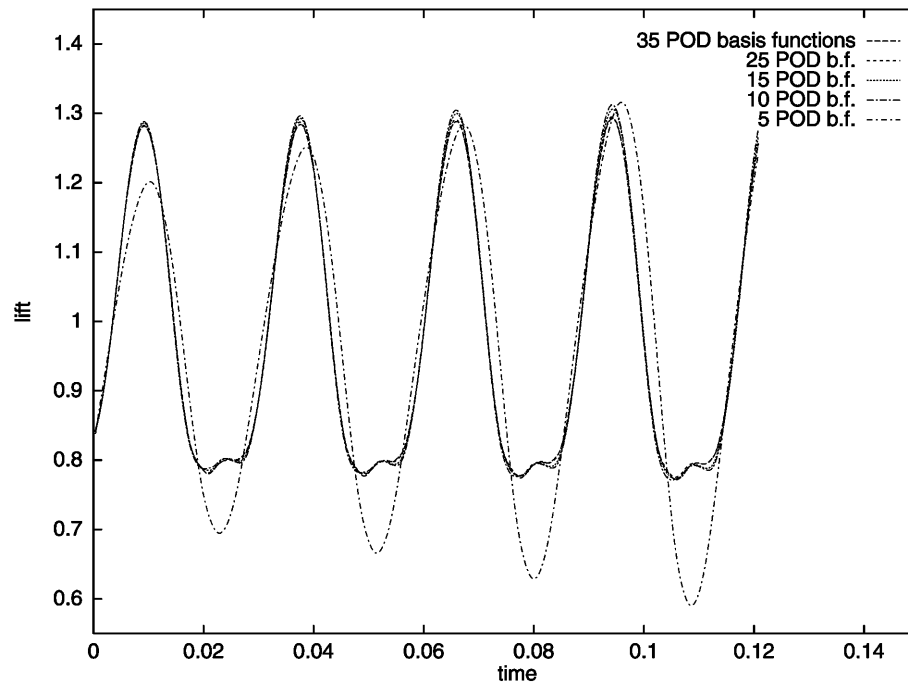


Figure 6. Comparison between the results obtained using the reference code and the ones obtained with a different number of POD basis functions at $CFL = 75$.

4.2. Constant Coefficients System of ODEs

Consider the Navier–Stokes equations rewritten in terms of the primitive variables $\{\varsigma, u, v, p\}$, where $\varsigma = 1/\rho$. Other more usual forms of these equations introduce rational fractions of the unknowns leading to a system of ODEs which is not polynomial, as, for example, in [16]. We have

$$\begin{aligned}
& \varsigma_t + u v_x - u_x \varsigma - v_y \varsigma + v \varsigma_y = 0, \\
& u_t + u u_x + v u_y + \varsigma p_x = \frac{1}{Re} \varsigma \left[\left(\frac{4}{3} u_x - \frac{2}{3} v_y \right)_x + (v_x + u_y)_y \right], \\
& v_t + u v_x + v v_y + \varsigma p_y = \frac{1}{Re} \varsigma \left[\left(\frac{4}{3} v_y - \frac{2}{3} u_x \right)_y + (v_x + u_y)_x \right], \\
& p_t + u p_x + v p_y + \gamma p (v_x + u_y) \\
& = \frac{\gamma}{Re Pr} [(p \varsigma)_{xx} + (p \varsigma)_{yy}] + \frac{\gamma - 1}{Re} \left[u_x \left(\frac{4}{3} u_x - \frac{2}{3} v_y \right) + v_y \left(\frac{4}{3} v_y - \frac{2}{3} u_x \right) + (u_y + v_x)^2 \right].
\end{aligned} \tag{19}$$

After projection over $\{\phi_1^i, \phi_2^i, \phi_3^i, \phi_4^i\}$, integrating by parts and discarding negligible boundary terms, we obtain a system of the form

$$\dot{w}_i + \sum_{j,k=1}^M a_{i,j,k} w_j w_k + \frac{1}{Re} \sum_{j,k=1}^M b_{i,j,k} w_j w_k = 0, \tag{20}$$

where

$$\begin{aligned}
a_{i,j,k} = & \int_{\Omega} \left(\phi_1^i \phi_2^j \frac{\partial \phi_1^k}{\partial x} + \phi_1^i \phi_3^j \frac{\partial \phi_1^k}{\partial y} - \phi_1^i \phi_1^k \frac{\partial \phi_2^j}{\partial x} - \phi_1^i \phi_1^k \frac{\partial \phi_3^j}{\partial y} \right) d\Omega \\
& + \int_{\Omega} \left(\phi_2^i \phi_2^j \frac{\partial \phi_2^k}{\partial x} + \phi_2^i \phi_3^j \frac{\partial \phi_2^k}{\partial y} + \phi_2^i \phi_1^j \frac{\partial \phi_4^k}{\partial x} \right) d\Omega \\
& + \int_{\Omega} \left(\phi_3^i \phi_2^j \frac{\partial \phi_3^k}{\partial x} + \phi_3^i \phi_3^j \frac{\partial \phi_3^k}{\partial y} + \phi_3^i \phi_1^j \frac{\partial \phi_4^k}{\partial y} \right) d\Omega \\
& + \int_{\Omega} \left(\phi_4^i \phi_2^j \frac{\partial \phi_4^k}{\partial x} + \phi_4^i \phi_3^j \frac{\partial \phi_4^k}{\partial y} + \gamma \phi_4^i \phi_4^j \frac{\partial \phi_2^k}{\partial x} + \gamma \phi_4^i \phi_4^j \frac{\partial \phi_3^k}{\partial y} \right) d\Omega
\end{aligned} \tag{21}$$

and

$$\begin{aligned}
b_{i,j,k} = & \int_{\Omega} \left(\frac{4}{3} \frac{\partial \phi_2^i}{\partial x} \phi_1^j \frac{\partial \phi_2^k}{\partial x} - \frac{2}{3} \frac{\partial \phi_2^i}{\partial x} \phi_1^j \frac{\partial \phi_3^k}{\partial y} \right) d\Omega + \int_{\Omega} \left(\frac{\partial \phi_3^i}{\partial x} \phi_1^j \frac{\partial \phi_2^k}{\partial y} + \frac{\partial \phi_3^i}{\partial x} \phi_1^j \frac{\partial \phi_3^k}{\partial x} \right) d\Omega \\
& + \frac{\gamma}{Pr} \int_{\Omega} \left(\frac{\partial \phi_4^i}{\partial x} \phi_1^j \frac{\partial \phi_4^k}{\partial x} + \frac{\partial \phi_4^i}{\partial x} \phi_4^j \frac{\partial \phi_1^k}{\partial x} \right) d\Omega + \int_{\Omega} \left(\frac{4}{3} \frac{\partial \phi_3^i}{\partial y} \phi_1^j \frac{\partial \phi_3^k}{\partial y} - \frac{2}{3} \frac{\partial \phi_3^i}{\partial y} \phi_1^j \frac{\partial \phi_2^k}{\partial x} \right) d\Omega \\
& + \int_{\Omega} \left(\frac{\partial \phi_2^i}{\partial y} \phi_1^j \frac{\partial \phi_2^k}{\partial y} + \frac{\partial \phi_2^i}{\partial y} \phi_1^j \frac{\partial \phi_3^k}{\partial x} \right) d\Omega + \frac{\gamma}{Pr} \int_{\Omega} \left(\frac{\partial \phi_4^i}{\partial y} \phi_1^j \frac{\partial \phi_4^k}{\partial y} + \frac{\partial \phi_4^i}{\partial y} \phi_4^j \frac{\partial \phi_1^k}{\partial y} \right) d\Omega \\
& + (\gamma - 1) \int_{\Omega} \left(\frac{4}{3} \frac{\partial \phi_2^j}{\partial x} \phi_4^i \frac{\partial \phi_2^k}{\partial x} - \frac{2}{3} \frac{\partial \phi_2^j}{\partial x} \phi_4^i \frac{\partial \phi_3^k}{\partial y} \right) d\Omega \\
& + (\gamma - 1) \int_{\Omega} \left(\frac{4}{3} \frac{\partial \phi_3^j}{\partial y} \phi_4^i \frac{\partial \phi_3^k}{\partial y} - \frac{2}{3} \frac{\partial \phi_3^j}{\partial y} \phi_4^i \frac{\partial \phi_2^k}{\partial x} \right) d\Omega \\
& + (\gamma - 1) \int_{\Omega} \left(\frac{\partial \phi_2^j}{\partial y} \phi_4^i \frac{\partial \phi_2^k}{\partial y} + 2 \frac{\partial \phi_3^j}{\partial x} \phi_4^i \frac{\partial \phi_2^k}{\partial y} + \frac{\partial \phi_3^j}{\partial x} \phi_4^i \frac{\partial \phi_3^k}{\partial x} \right) d\Omega.
\end{aligned} \tag{22}$$

The coefficients $a_{i,j,k}$ and $b_{i,j,k}$ are computed once and for all using the basis functions and their gradients computed as in (11). The system of ODEs is integrated in time by the backward differentiation formula for stiff systems of ODEs (Gear method).

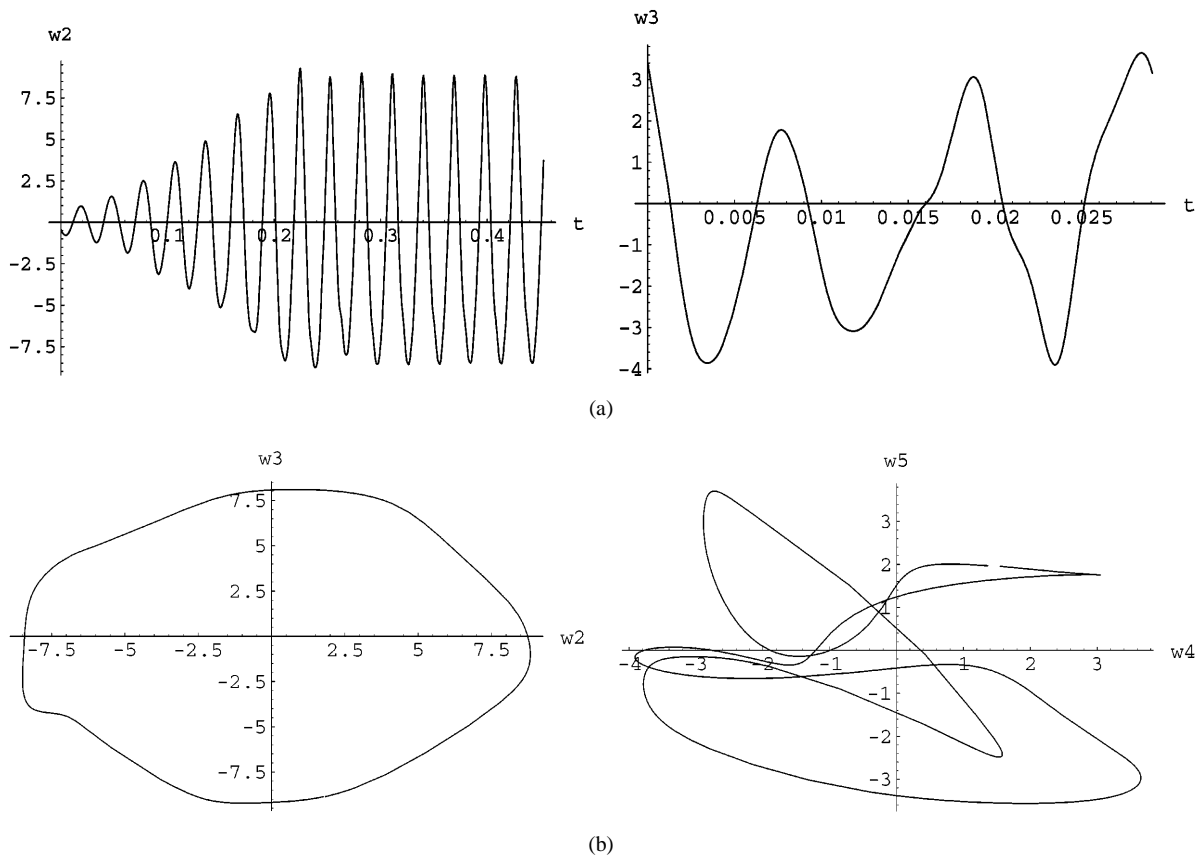


Figure 7. Constant coefficient system without additional dissipation: (a) second and third coefficient versus time, (b) phase-space sections.

In the airfoil test case, the period (0.028 s) of the oscillations found by numerical integration of (19) is in agreement with that obtained using the reference or the POD filter scheme. Yet, one can see in Figure 7(a) for the coefficients $w_2(t)$ and $w_3(t)$ that amplitude of the oscillation increases finally to stabilize. In the same figure the phase-space cut on the plane of the coefficients w_2-w_3 and w_4-w_5 is compared with what is obtained by POD filtering. The ordered motion of Figure 8 is disrupted in Figure 7(b). We ascribe this behavior to a lack of numerical diffusion. If additional stabilization is provided in a brutal way, i.e., increasing the viscosity, the results of the constant coefficients system are in good agreement with those of the reference numerical simulation, compare Figure 8 and Figure 9(b).

We now turn our attention to a turbulent test case: we consider a square cylinder immersed in flow at Mach number 0.1 and Reynolds number 22 000. A $\kappa-\varepsilon$ model with wall laws was applied. The flow field is discretized over a triangular mesh with 9274 vertices. The flow computation performed with the reference finite-volumes code shows a periodic vortex shedding from the corners of the square. In order to build the basis functions we took 100 snapshots of the flow over one period; the first component and the third component of $\phi^2-\phi^8$ are shown in Figures 11 and 10. We observe an odd-even decoupling of the modes as well as a phase shift, as in the previous case.

The POD filter was constructed with 30 basis functions and the results are shown in Figure 12. Again the low order model is in good agreement with the reference code. The phase-space sections for w_2-w_3 , w_4-w_5 and w_6-w_7 are given in Figure 13. It is seen that the motion on this plane is ordered and that the shapes of the closed orbits denote the presence of oscillation with increasing time frequency as the number of basis functions increases.

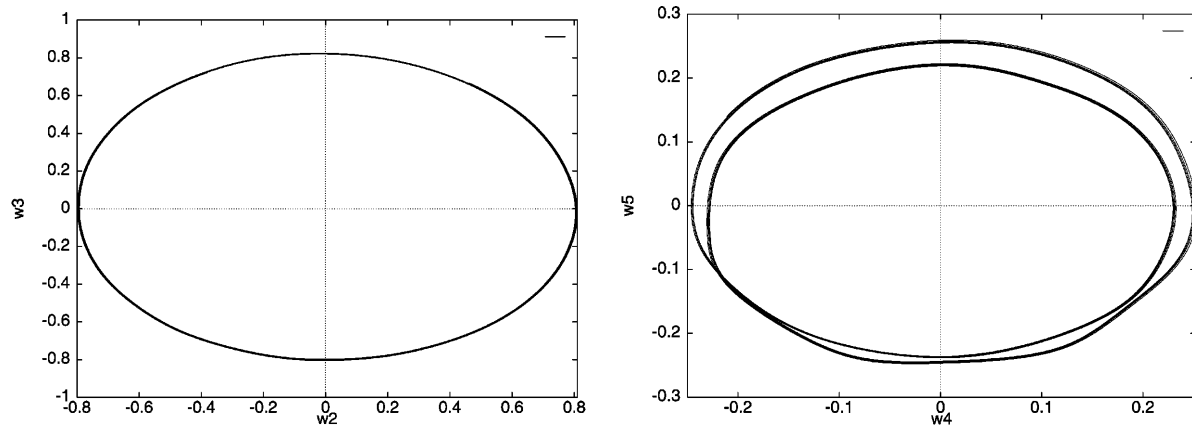


Figure 8. Projection of the numerical solution obtained by the FV code: phase–space sections.

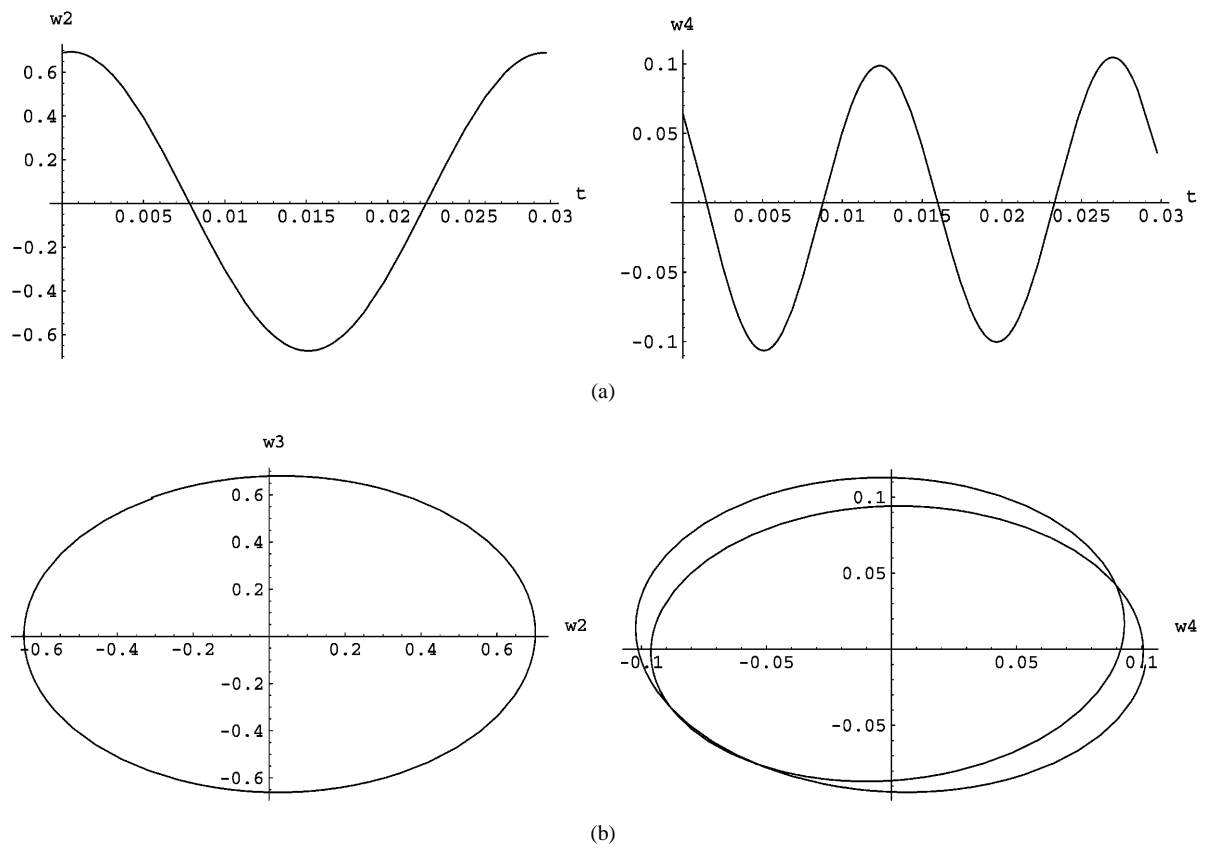


Figure 9. System with increased viscosity: (a) second and third coefficient versus time, (b) phase–space sections.

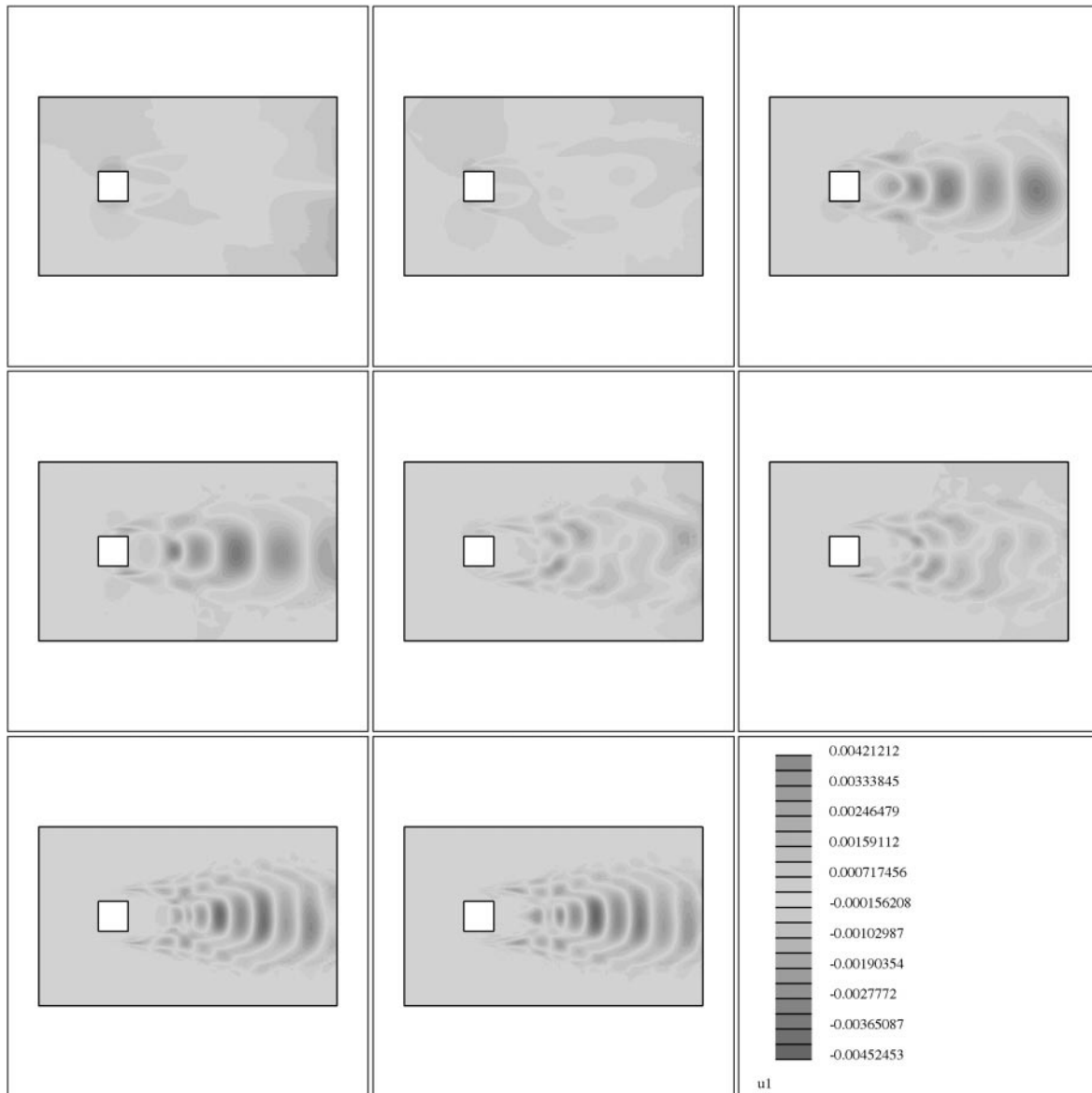


Figure 10. Square cylinder, Mach = 0.1, $Re = 22\,000$, turbulent. 100 Snapshots; 10 basis functions. First component of POD basis functions.

4.3. H^1 Formulation

In the POD-filter code we used for the laminar and turbulent flow experiments, the numerical diffusion is introduced by solving Riemann problems at control volume interfaces, ensuring sufficient stabilization so that the low order model is also stable. In order to develop a rational method to introduce diffusion in the constant coefficient system of ODEs (19), we explored the possibility of redefining the norms involved in the POD definition of Section 2. Indeed, it seems to us that to increase the relevance of smaller scales in the POD definition might enhance dissipation. This can be accomplished, for example, by redefining all the norms in the Sobolev space H^1 , so that the derivatives of the snapshots as well as that of the basis functions are included in the POD average. The formula to compute the POD basis functions becomes

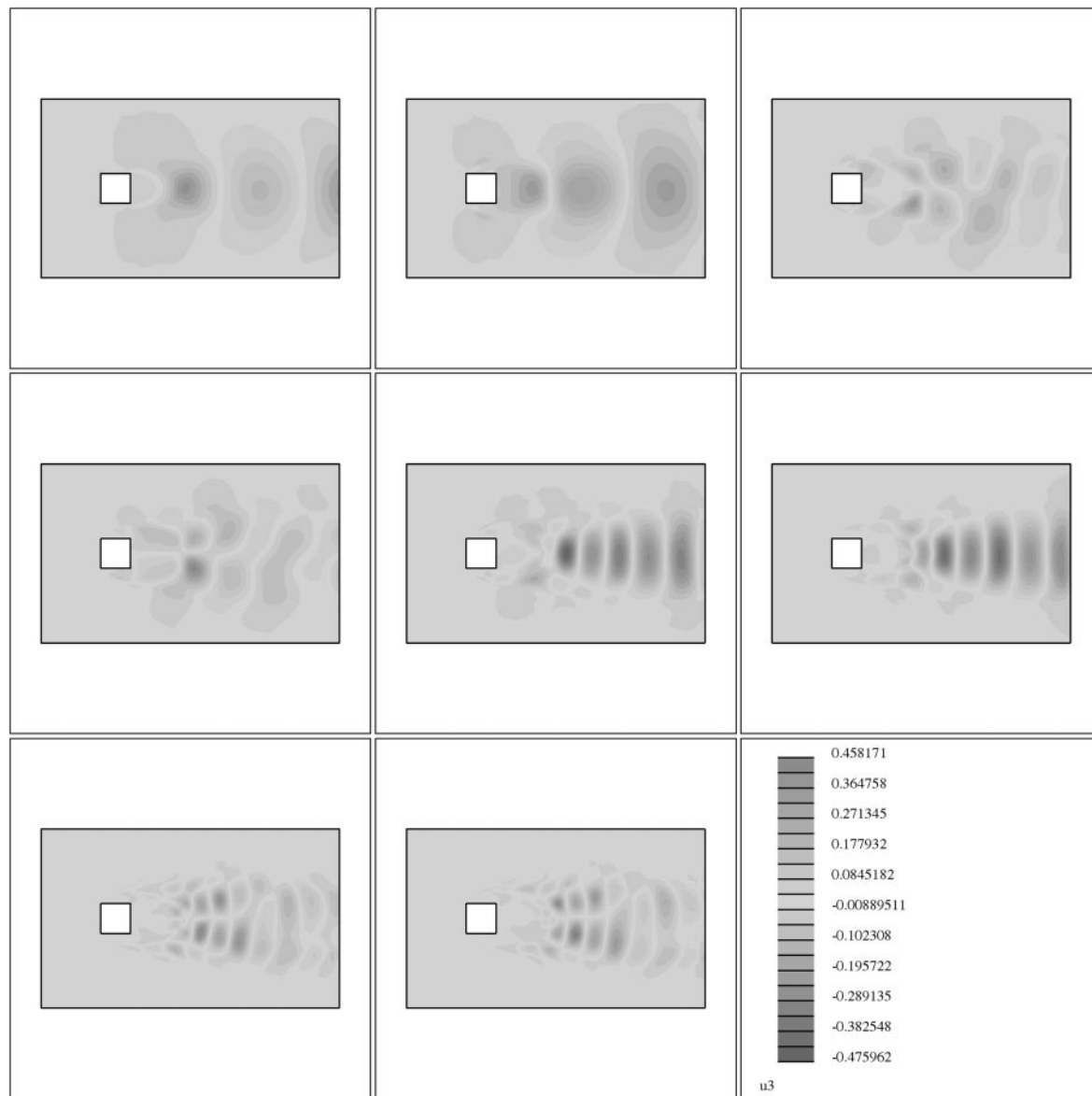


Figure 11. Third component of POD basis functions for the square cylinder case.

$$\begin{aligned}
 \phi &= \max_{\psi \in H^1} \langle (W, \psi)^2 \rangle \\
 &= \max_{\psi \in H^1} \left\langle \left(\int_{\Omega} W \psi \, d\Omega + \varepsilon \int_{\Omega} \nabla W \cdot \nabla \psi \, d\Omega \right)^2 \right\rangle.
 \end{aligned} \tag{23}$$

Many metrics are possible for H^1 , so that ε can be conveniently chosen in different simulations. From dimensional analysis considerations, one may guess ε proportional to T/Re , where T is some appropriate time scale. The basis functions are found by linear combination of the solution snapshots as in the L^2 case, yet the coefficients of the linear combination are now the eigenvectors of the matrix

$$\{c_{nm}\} = \left\{ \int_{\Omega} W^{(n)} W^{(m)} \, d\Omega + \varepsilon \left(\int_{\Omega} \nabla W^{(n)} \cdot \nabla W^{(m)} \, d\Omega \right) \right\}, \tag{24}$$

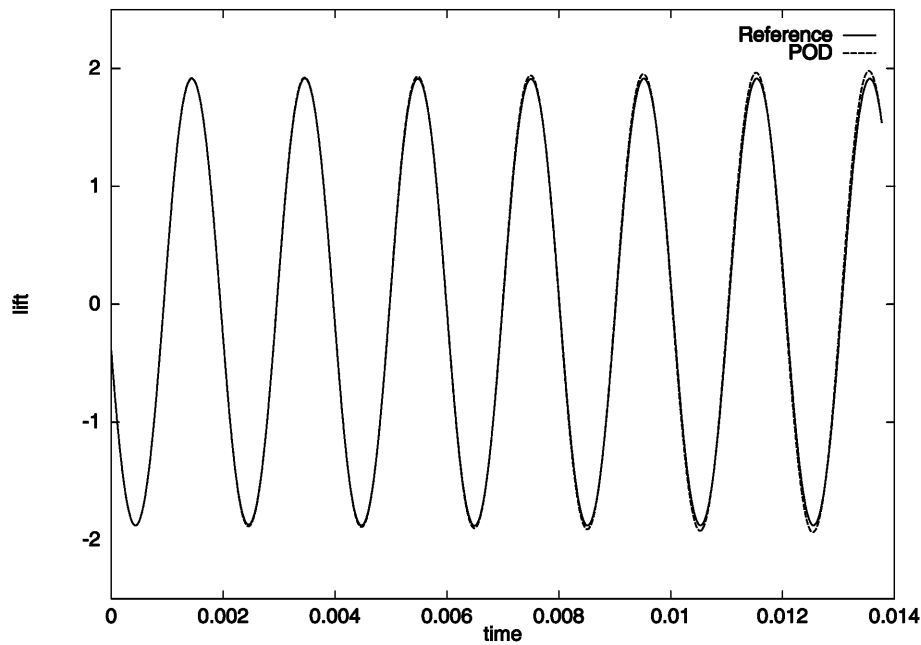


Figure 12. Results of the reference code compared with the ones obtained by POD filtering in the turbulent case.

where the numerical evaluation of the nodal gradient ∇W is computed as an average over the cell of the P1 gradients. Accordingly, the orthogonality condition is $(\phi^i, \phi^j)_{H^1} = \delta_{ij}$.

The numerical simulations which we now present are obtained based on the POD-filter code in which the filter is built using the basis defined in H^1 . Our aim is to investigate the approximation properties of such functions compared with those obtained with the L^2 norm.

We expect that as the Reynolds number is increasing, the advantage of the H^1 formulation over the L^2 becomes more and more evident. In the laminar test case, which was studied for $Re = 2100$, we now double the Reynolds number and compute the flow starting from a uniform initial condition with the basis functions relative to the $Re = 2100$ case. This simulation is different from that of Figure 5, as the transitory solution occurring before the establishment of the periodic vortex shedding does not belong to the database from which we derive the POD basis functions. Nevertheless, it is seen in Figure 14(a) that POD basis functions derived with the H^1 norm are still in good agreement with the reference solution, whereas the L^2 -norm basis functions are unable to give a reasonable representation of the flow. If we initialize the computation from the established periodic flow, still at $Re = 4200$, the solution obtained with the L^2 -norm basis functions compare better with the reference flow, see Figure 14(b). Yet, it is seen that the amplitude of the oscillation tends to increase with time.

Indeed, the next example also shows that the H^1 basis functions seem to allow better results even in the case of a turbulent flow. The experimental setup is the following: in the cylinder case ($Re = 22000$) the convective fluxes are computed using a centered approximation. Then we compare the L^2 POD filtered solution and the H^1 POD filtered solution. The reference code is in this case numerically unstable, as well as the L^2 POD filter code. On the contrary, the H^1 formulation with $\varepsilon = 10$ and ten basis functions provides sufficient stabilization so that the computation is still in good agreement with the reference computation, as is seen in Figure 15 where it is shown how the curve corresponding to the L^2 basis (solid line) tends to diverge from the reference, whereas the curve relative to the H^1 basis (dashed line) has constant amplitude over six periods. Therefore, we observe a definite benefit in employing the H^1 formulation of the POD.

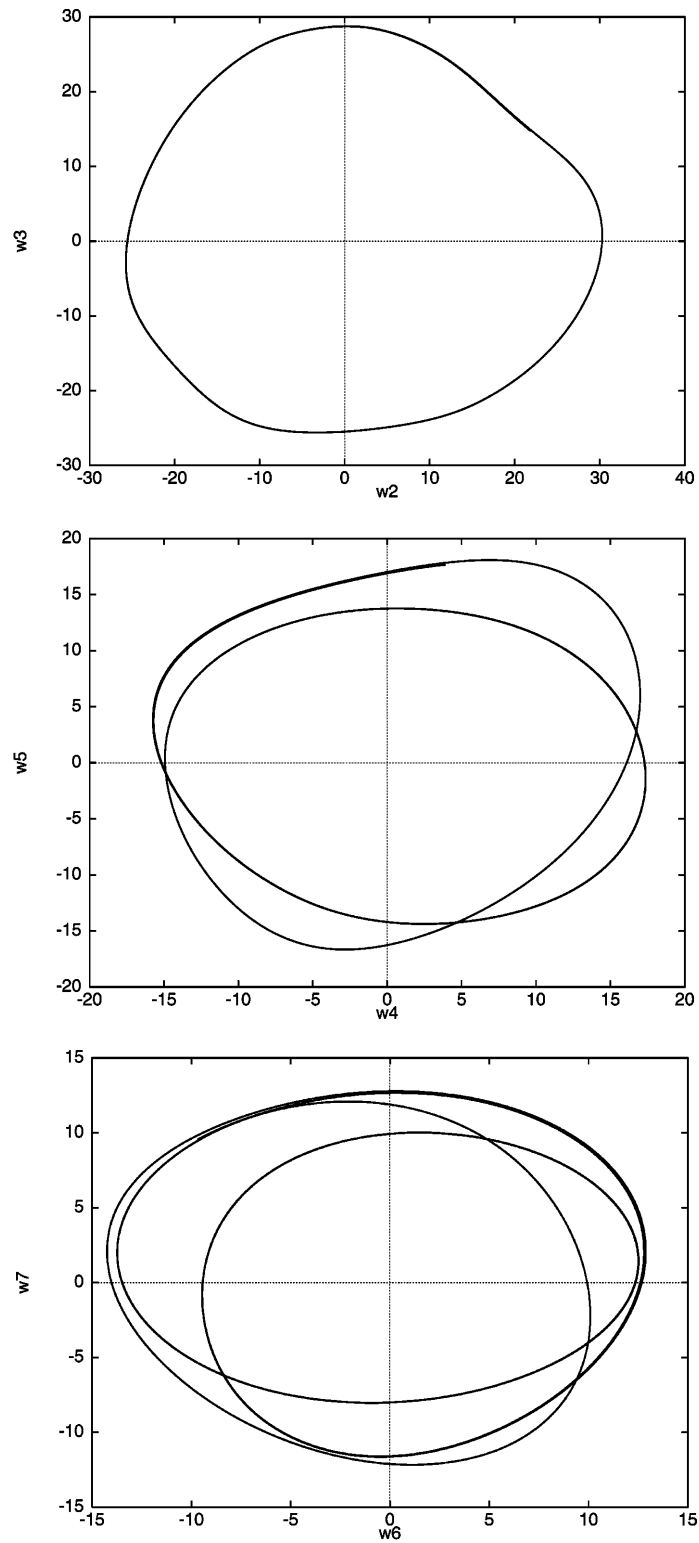
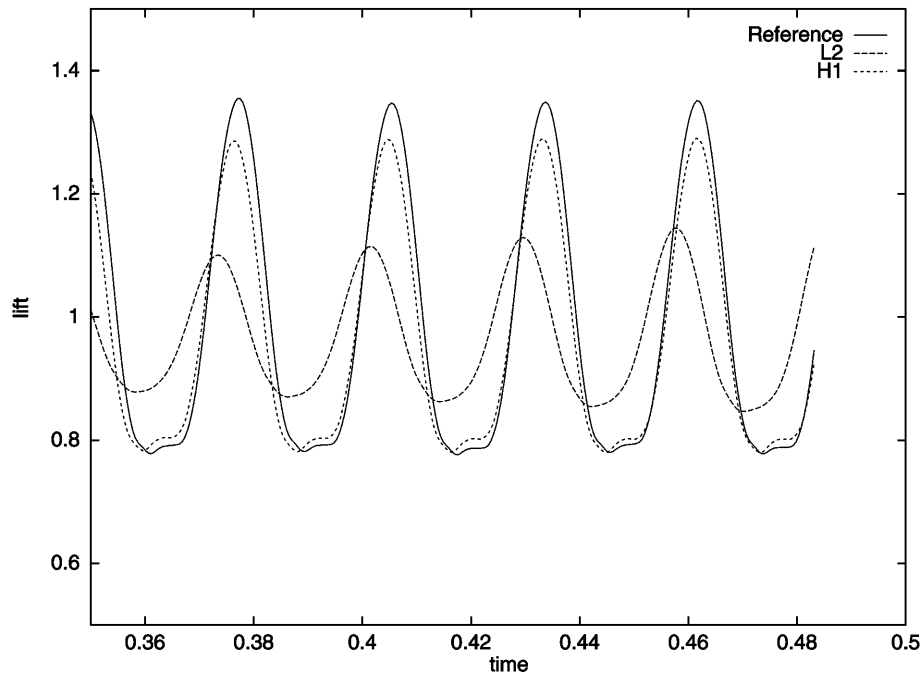
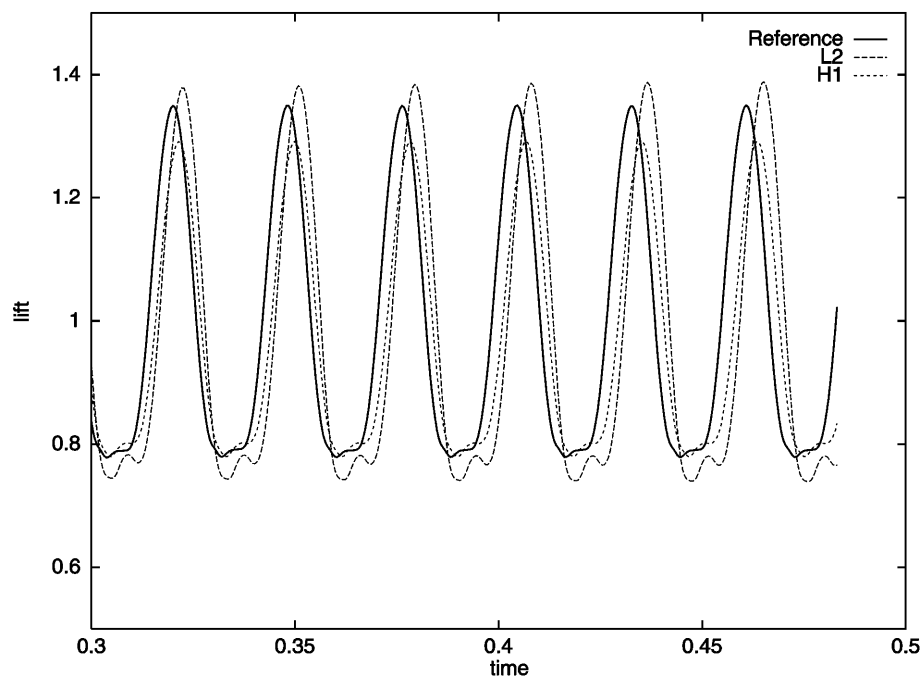


Figure 13. Square cylinder: phase space sections.



(a)



(b)

Figure 14. $Re = 4200$. Comparison between the results obtained using the reference code and the ones obtained by POD filtering with basis functions relative to $Re = 2100$: (a) uniform flow initial condition; (b) initialization from the established periodic flow.

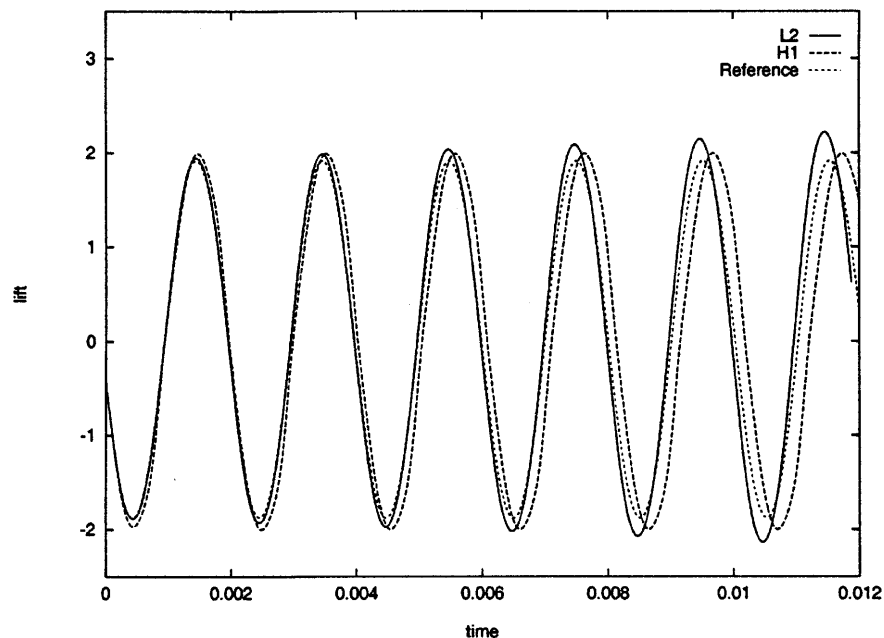


Figure 15. Square cylinder case: reference versus the L^2 and the H^1 POD-filter computations.

5. Conclusions

For the test cases we considered (airfoil at $Re = 2100$, $Mach = 0.2$, and square cylinder at $Re = 22\,000$, $Mach = 0.1$), finite-volumes codes need some kind of numerical stabilization. Indeed, if the convective fluxes are computed by central approximations, the numerical residuals diverge. It is well known that this occurs when the convective terms becomes preponderant over the diffusive terms.

The same way, as shown in the model problem of Section 3, POD–Galerkin approximations for convective dominated problems need numerical stabilization, as they amount, for example, to central or upwind schemes depending on the spatial differentiation operator used for the derivative of the POD basis functions.

We have shown that POD can be viewed as a filtering process. The results based on a POD-filter canonical code are in good agreement with the reference, both in a laminar and a turbulent case. We conclude that the expectation of simulating a complex flow on the basis of a limited number of degrees of freedom is well founded. Through the canonical code on which we base our POD approach we are able to provide a rational way to add numerical diffusion to the ODE system. We have derived the constant coefficients system of ODEs relative to the compressible Navier–Stokes equations, and shown that, however, its judicious numerical stabilization is crucial to retain the approximation properties found using the POD-filter code. Based on numerical experiments, we suggest that one way to achieve stabilization improvements is to define POD in the H^1 Sobolev space, that is to incorporate gradients as well as function values in the definition of POD. It turns out that taking into account derivatives in the definition of POD modes or representing the actions of small scales through an “eddy” viscosity as is done in [2] are essentially two different ways to accomplish the same result: recover the action of the small scales. In conclusion the perspectives opened are to devise numerical expedients to include the optimal diffusion amount in the constant system of ODEs relative to the compressible Navier–Stokes equations, taking advantage of the apparent benefits of the H^1 formulation.

Acknowledgments

The first author expresses his gratitude to Dr. A. Dervieux (INRIA) for his constant support and encouragement. Particular thanks are due to Dr. R. Peyret (CNRS) for valuable recommendations as well as to Dr. M. Mallet and Mr. G. Vigo (Dassault Aviation) for several fruitful discussions contributing to this work.

References

- [1] Aubry, N., Guyonnet, R., and Lima, R. (1992). Spatio-temporal symmetries and bifurcations via bi-orthogonal decompositions. *J. Nonlinear Sci.*, **2**, 183–215.
- [2] Aubry, N., Holmes, P., Lumley, J.L., and Stone, E. (1988). The dynamics of coherent structures in the wall region of a turbulent boundary layer. *J. Fluid Mech.*, **192**, 115–173.
- [3] Aubry, N., Lian, W.Y., and Titi, E.S. (1993). Preserving symmetries in the proper orthogonal decomposition. *SIAM J. Sci. Comput.*, **14**(2), 483–505.
- [4] Berkooz, G., Holmes, P., and Lumley, J.L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Ann. Rev. Fluid Mech.*, **25**, 539–575.
- [5] Berkooz, G., Holmes, P., and Lumley, J.L. (1993). On the relation between low-dimensional models and the dynamics of coherent structures in the turbulent wall layer. *Theo. Comput. Fluid Dynamics*, **4**, 255–269.
- [6] Courant, R., and Hilbert, D. (1953). *Methods of Mathematical Physics*, vol. I. Wiley, New York.
- [7] Iollo, A. (1997). Remarks on the Approximation of the Euler Equations by a Low Order Model. INRIA Research Report RR-3329.
- [8] Lumley, J.L. (1967). The structure of inhomogeneous turbulent flow. In *Atmospheric Turbulence and Wave Propagation*, (A.M. Yaglom and V.I. Tatarski, eds.), 166–178. Nauka, Moscow.
- [9] Martin, R., and Guillard, H. (1996). A second order defect correction scheme for unsteady problems. *Comput. Fluids*, **25**(1), 9–27.
- [10] Pulliam, T.H. (1989). Low Reynolds Number Numerical Solution of Chaotic Flows. AIAA Paper 89-0123.
- [11] Roe, P.L. (1981). Approximate Riemann solvers, parameters vectors and difference schemes. *J. Comput. Phys.*, **43**, 357–371.
- [12] Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. Part I: coherent structures. *Quart. Appl. Math.*, **XLV**(3), 561–571.
- [13] Tang, K.Y., Graham, W.R., and Peraire, J. (1996). Active Flow Control Using a Reduced Order Model and Optimum Control. AIAA Paper 96-1946.
- [14] Tran, H., Koobus, B., and Farhat, C. (1998). Numerical Solution of Vortex Dominated Flow Problems with Moving Grids. AIAA Paper 98-0766.
- [15] van Leer, B. (1979). Towards the ultimate conservative difference scheme, V: a second-order sequel to Godunov's method, *J. Comput. Phys.*, **32**, 361–370.
- [16] Vigo, G. (1998). The Proper Orthogonal Decomposition Applied to Unsteady Compressible Navier–Stokes Equations. INRIA Research Report RR-3385.