A POD-GALERKIN REDUCED MODEL WITH UPDATED COEFFICIENTS FOR SMAGORINSKY LES

Sebastian Ullmann*, **Jens Lang**
*ullmann@mathematik.tu-darmstadt.de

TECHNISCHE UNIVERSITÄT DARMSTADT

Research Group
Numerical Analysis and
Scientific Computing

Abstract

A novel Galerkin reduced order model based on proper orthogonal decomposition (POD) has been developed for large-eddy simulations (LES) with the Smagorinsky subgrid-scale model. We propose to use the subgrid-scale model of the original simulation to stabilize a POD reduced model based on LES snapshots in a consistent way. The varying eddy viscosity field can be taken into account in the reduced model by dynamically updating its coefficients. For the validation of the model, snapshots of an LES of the turbulent flow around a circular cylinder at $Re = 3900$ are used. It is shown that the reduced model reproduces the drag and lift coefficients reasonably and captures the kinetic energy spectrum well for low and medium frequencies, while the computational cost is significantly reduced.

1 INTRODUCTION

The possibility to approximate complex three-dimensional flows by low-dimensional models has gained attention by theoreticians, who hope to better understand the nature of flow phenomena, as well as practitioners, who hope to speed up their optimization or control processes. Research has focussed on reduced models resulting from a Galerkin projection of the flow equations on the coherent structures of the flow field, see Holmes et al. (1996) for details. Typically, the coherent structures are extracted from a set of representative flow snapshots by means of the snapshot POD introduced by Sirovich (1987). Although it is often possible to capture a large fraction of the kinetic energy of the flow field by a linear combination of a small number of POD modes, robustness is still an issue in POD-Galerkin reduced modeling. Spurious limit cycles and even divergence have been observed already for simple flow configurations at low Reynolds numbers (Bergmann et al., 2009).

Several improvements have been suggested and validated recently, but it is still a challenge to progress to turbulent flows, where the larger coherent structures are superposed by an increasing fraction of chaotic flow details. We mention a few publications that focus on reduced models for transitional and turbulent flows: Telib et al. (2004) used the least-squares calibration method introduced by Galletti et al. (2004) to build a reduced model of the flow in a T-mixer, which gave good results for the transitional flow at $Re = 300$ and $Re = 400$. Buffoni et al. (2006) applied the method of Galletti et al. (2007), which is based on a pseudo-spectral model calibration, to the three-dimensional flow around a square cylinder at $Re = 300$. Couplet et al. (2005) described a calibrated reduced-order model for the turbulent flow past a backward-facing step based on snapshots generated by a large-eddy simulation. They chose a Reynolds number of 7432 based on the step height. The authors briefly mentioned the possibility of taking an LES subgrid-scale model into account in the Galerkin model, but rejected this approach in favor of calibrating the coefficients of a reduced model based on the Navier-Stokes equations.

In this work we pick up the thought of incorporating a subgrid-scale model as a stabilization technique for the POD-Galerkin modeling of turbulent flows. We take the LES equations with the Smagorinsky subgrid-scale model and perform a Galerkin projection on the POD modes. Since the eddy viscosity introduced by Smagorinsky (1963) is highly non-linear, we do not take it into account in the Galerkin projection, but update the corresponding model coefficients during the time integration of the reduced model.

We test the presented approach using LES snapshots of the flow around a circular cylinder at $Re = 3900$. The simulations are performed with the code KARDOS (Erdmann et al., 2002), which incorporates a stabilized Galerkin/least–squares finite element method. The equations are discretized in space with linear finite elements on a tetrahedral mesh and in time with the Rosenbrock method ROS3P using automatic step size control; see Lang (1998) for details.

2 REDUCED-ORDER MODEL

We are given a tetrahedral finite element mesh consisting of the mesh nodes $\mathbf{x}_1, \ldots, \mathbf{x}_M$. At every snapshot time t_n, $n = 1, \ldots, N$, and at every mesh node $\mathbf{x}_m$ velocity values u^i_{mn}, $i = 1, 2, 3$, are available from a three-dimensional direct or large-eddy simulation. The snapshot ensemble mean of the velocity at each

nodal point is given by $\bar{u}_m^i = \frac{1}{N}\sum_{n=1}^{N} u_{mn}^i$. Using the finite element functions $\psi_m(\mathbf{x})$ with local support, the velocities and the mean velocities can be expressed as functions of space in the following way:

$$u_n^i(\mathbf{x}) = \sum_{m=1}^{M} u_{mn}^i \psi_m(\mathbf{x}), \quad n = 1,\dots,N, \quad i = 1,2,3,$$
$$\bar{u}_i(\mathbf{x}) = \sum_{m=1}^{M} \bar{u}_m^i \psi_m(\mathbf{x}), \quad i = 1,2,3.$$

We define a snapshot matrix S by

$$S = \begin{pmatrix} WS^1 \\ WS^2 \\ WS^3 \end{pmatrix}, \quad \text{where} \quad S^i = \begin{pmatrix} u_{11}^i - \bar{u}_1^i & \dots & u_{1N}^i - \bar{u}_1^i \\ \vdots & & \vdots \\ u_{M1}^i - \bar{u}_M^i & \dots & u_{MN}^i - \bar{u}_M^i \end{pmatrix}$$

and W is a suitable weighting matrix, e.g. the identity matrix or a matrix resulting from the spatial discretization of the problem; see Kunisch and Volkwein (1999). Note that we use the velocity fluctuations around the snapshot mean to set up the snapshot matrix. The truncated singular value decomposition of this snapshot matrix yields

$$S \approx U_R \Sigma_R V_R^T = \begin{pmatrix} WU_R^1 \\ WU_R^2 \\ WU_R^3 \end{pmatrix} \Sigma_R V_R^T \quad \text{with} \quad U_R^i = \begin{pmatrix} \varphi_{11}^i & \dots & \varphi_{1R}^i \\ \vdots & & \vdots \\ \varphi_{M1}^i & \dots & \varphi_{MR}^i \end{pmatrix},$$

where R is the rank of the truncation. The matrix Σ_R is a diagonal matrix with the singular values σ_r, ordered decreasingly by magnitude, on the diagonal. The squares of the singular values are called POD energies, denoted by $\lambda_r = \sigma_r^2$. Using the elements of the orthogonal matrix U_R in combination with their respective finite element functions, we define the POD modes of the velocity components as

$$\varphi_r^i(\mathbf{x}) = \sum_{m=1}^{M} \varphi_{mr}^i \psi_m(\mathbf{x}), \quad r = 1,\dots,R, \quad i = 1,2,3.$$

With an adequate number of representative snapshots of a flow field that lends itself reasonably well to a POD, and with a sufficiently large R, we are able to approximate any realization of the flow field by a linear combination of the snapshot mean and the POD modes:

$$u_i(\mathbf{x},t) = \bar{u}_i(\mathbf{x}) + \sum_{r=1}^{R} a_r(t)\varphi_r^i(\mathbf{x}), \quad i = 1,2,3. \tag{1}$$

The continuity equation is automatically fulfilled by the velocity POD, provided that the snapshots are divergence free. The Navier-Stokes momentum equations are given by

$$\partial_t u_i + u_j \partial_{x_j} u_i + \partial_{x_i} p = \partial_{x_j}(\nu(\partial_{x_j} u_i + \partial_{x_i} u_j)), \quad i = 1,2,3,$$

using Einstein summation over the index j. By a Galerkin projection on the velocity POD (1) we obtain a system of ordinary differential equations

$$A_{qr}\dot{a}_q = B_{pqr} a_p a_q + C_{qr} a_q + D_r, \quad r = 1,\dots,R \tag{2}$$

for the POD coefficients $a_1(t),\dots,a_R(t)$. Here we use the Einstein summation convention to sum over p and q, taking values from 1 to R.

If the positive square root of the finite element mass matrix is used as a weighting matrix W for the POD, then the matrix formed by the coefficients A_{qr} in (2) simplifies to the identity matrix. This, however, does hardly influence the computational effort of a numerical time integration scheme, since the evaluation of the right-hand side is more expensive than the forward and backward substitution necessary if the equations are coupled.

The Galerkin projection on the divergence-free POD modes and subsequent integration by parts reduces the pressure term to a boundary integral that nearly vanishes. According to the discussion in Noack et al. (2005) it is not always possible to neglect the pressure, but its importance reduces with an increase of the Reynolds number. In this study we do not take the pressure into account for the time integration of the reduced model.

We define the LES momentum equations with the Smagorinsky subgrid-scale model as

$$\partial_t u_i + u_j \partial_{x_j} u_i + \partial_{x_i} p = \partial_{x_j}((\nu + \nu^{\mathrm{t}}/2)(\partial_{x_j} u_i + \partial_{x_i} u_j)), \quad i = 1,2,3,$$

where

$$\nu^{\mathrm{t}} = (C_{\mathrm{S}}\Delta)^2 \sqrt{\frac{\partial_{x_i} u_j \partial_{x_i} u_j}{2} + \frac{\partial_{x_i} u_j \partial_{x_j} u_i}{2}}, \tag{3}$$

with the Smagorinsky constant C_{S} and with the local filter width Δ corresponding to the maximum edge length of each tetrahedron. We perform a Galerkin projection of the LES momentum equations on the POD, leaving ν^{t} as a time- and space-dependent parameter. The resulting system of ordinary differential equations is given by

$$A_{qr}\dot{a}_q = B_{pqr} a_p a_q + (C_{qr} + C^{\mathrm{t}}_{qr}(t)) a_q + D_r + D^{\mathrm{t}}_r(t), \quad r = 1,\ldots,R, \tag{4}$$

where for any q and r

$$C^{\mathrm{t}}_{qr}(t) = \left(\varphi^i_r(\mathbf{x}), \partial_{x_j}\nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_j}\varphi^i_q(\mathbf{x})\right) + \left(\varphi^i_r(\mathbf{x}), \partial_{x_j}\nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_i}\varphi^j_q(\mathbf{x})\right), \tag{5}$$

$$D^{\mathrm{t}}_r(t) = \left(\varphi^i_r(\mathbf{x}), \partial_{x_j}\nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_j}\bar{u}_i(\mathbf{x})\right) + \left(\varphi^i_r(\mathbf{x}), \partial_{x_j}\nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_i}\bar{u}_j(\mathbf{x})\right). \tag{6}$$

To compute C^{t}_{qr} and D^{t}_r we first construct the derivatives of the velocity field from the current POD coefficients using (1) together with suitable discrete derivative operators. For linear finite elements the derivatives of the velocities and POD modes are constant within each tetrahedron. Using (3) we can compute the turbulent eddy viscosity ν^{t}. After integrating (5) and (6) by parts and neglecting the boundary terms we approximate the coefficients by

$$C^{\mathrm{t}}_{qr}(t) \approx \tilde{C}^{\mathrm{t}}_{qr}(t) = -\left(\partial_{x_j}\varphi^i_r(\mathbf{x}), \nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_j}\varphi^i_q(\mathbf{x})\right) - \left(\partial_{x_j}\varphi^i_r(\mathbf{x}), \nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_i}\varphi^j_q(\mathbf{x})\right),$$
$$D^{\mathrm{t}}_r(t) \approx \tilde{D}^{\mathrm{t}}_r(t) = -\left(\partial_{x_j}\varphi^i_r(\mathbf{x}), \nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_j}\bar{u}_i(\mathbf{x})\right) - \left(\partial_{x_j}\varphi^i_r(\mathbf{x}), \nu^{\mathrm{t}}(\mathbf{x},t)\partial_{x_i}\bar{u}_j(\mathbf{x})\right).$$

Let V_m be the volume of the tetrahedron with index m and let $\mathbf{x}_m$ be some location inside this tetrahedron. Assuming linear finite elements we can now write

$$\tilde{C}^{\mathrm{t}}_{qr}(t) = -\sum_{m=1}^{M_{\mathrm{tetra}}} V_m \left(\partial_{x_j}\varphi^i_r(\mathbf{x}_m)\partial_{x_j}\varphi^i_q(\mathbf{x}_m) + \partial_{x_j}\varphi^i_r(\mathbf{x}_m)\partial_{x_i}\varphi^j_q(\mathbf{x}_m)\right)\nu^{\mathrm{t}}(\mathbf{x}_m,t),$$
$$\tilde{D}^{\mathrm{t}}_r(t) = -\sum_{m=1}^{M_{\mathrm{tetra}}} V_m \left(\partial_{x_j}\varphi^i_r(\mathbf{x}_m)\partial_{x_j}\bar{u}_i(\mathbf{x}_m) + \partial_{x_j}\varphi^i_r(\mathbf{x}_m)\partial_{x_i}\bar{u}_j(\mathbf{x}_m)\right)\nu^{\mathrm{t}}(\mathbf{x}_m,t),$$

where M_{tetra} is the total number of tetrahedra. We see that the updates of the coefficients are inner products of the time-dependent eddy viscosity vector with some time-constant data that can be preprocessed and stored as $R^2 + R$ vectors in memory.

With suitable initial conditions at hand we can solve the system of equations (4) using a numerical time integrator, which requires several evaluations of the right-hand side of the system during the solution process. The computationally most expensive part of the evaluation of the right-hand side is the updating of the time-dependent coefficients. To save computing time, we can extrapolate the coefficients $\tilde{C}^{\mathrm{t}}_{qr}$ and $\tilde{D}^{\mathrm{t}}_{r}$ over some steps of the time integrator. This procedure is described in the following: Let the sequence of time steps of the time integrator be $t_1 < t_2 < \cdots < t_N$. From t_1 to t_4 we compute the coefficients in all steps. We define the quadratic polynomials $c^{1,2,3}_{qr}(t) = c(\tilde{C}^{\mathrm{t}}_{qr}(t_1), \tilde{C}^{\mathrm{t}}_{qr}(t_2), \tilde{C}^{\mathrm{t}}_{qr}(t_3), t)$ and $d^{1,2,3}_{r}(t) = d(\tilde{D}^{\mathrm{t}}_{r}(t_1), \tilde{D}^{\mathrm{t}}_{r}(t_2), \tilde{D}^{\mathrm{t}}_{r}(t_3), t)$ that interpolate the values at the first three time steps and measure the relative extrapolation errors at t_4 using

$$e_{C^{\mathrm{t}}} = \Big(\sum_{q,r=1}^{R} \Big(c^{1,2,3}_{qr}(t_4) - C^{\mathrm{t}}_{qr}(t_4) \Big)^2 \Big)^{\frac{1}{2}} / \Big(\sum_{q,r=1}^{R} C^{\mathrm{t}}_{qr}(t_4)^2 \Big)^{\frac{1}{2}},$$
$$e_{D^{\mathrm{t}}} = \Big(\sum_{r=1}^{R} \Big(d^{1,2,3}_{r}(t_4) - D^{\mathrm{t}}_{r}(t_4) \Big)^2 \Big)^{\frac{1}{2}} / \Big(\sum_{r=1}^{R} D^{\mathrm{t}}_{r}(t_4)^2 \Big)^{\frac{1}{2}}.$$

Given some error tolerance ϵ, the times for the next required updates are defined as

$$t_{C^{\mathrm{t}}} = t_4 + \min \Big(2 \cdot (t_4 - t_3), \big(\epsilon / e_{C^{\mathrm{t}}}(t_4)\big)^{\frac{1}{3}} \cdot (t_4 - t_3) \Big),$$
$$t_{D^{\mathrm{t}}} = t_4 + \min \Big(2 \cdot (t_4 - t_3), \big(\epsilon / e_{D^{\mathrm{t}}}(t_4)\big)^{\frac{1}{3}} \cdot (t_4 - t_3) \Big).$$

At the time points $t_n, n > 4$, we take $c^{2,3,4}_{qr}(t_n)$ as values for the model coefficients as long as $t_n + (t_n - t_{n-1}) < t_{C^{\mathrm{t}}}$. Otherwise, we perform an update, define $c^{3,4,n}_{qr}(t)$ as the new extrapolation polynomial, compute the next update time and continue the time integration. Independently, we control the updates of D^{t}_{r} in the same manner.

3 RESULTS

We set up two test cases of flows around a circular cylinder. The first simulation of a laminar periodic flow at $Re = 100$ is used to validate the standard model (2) without eddy viscosity. The second simulation of a turbulent flow at $Re = 3900$ is used to test the stability, accuracy and efficiency of the reduced LES model (4).

3.1 Periodic flow

We run a direct numerical simulation (DNS) with a Reynolds number of $Re = 100$ based on the cylinder diameter, the inflow velocity and the kinematic viscosity. The cylinder of diameter $d = 1$ is centered at $x_1 = x_2 = 2$ aligned with the x_3-axis within a box-shaped domain of $0 \leq x_1 \leq 10$, $0 \leq x_2 \leq 4$ and $0 \leq x_3 \leq 0.1$. This is a pseudo two-dimensional setting, which corresponds to the two-dimensional nature of the flow around a circular cylinder in the laminar vortex shedding regime typical for the chosen Reynolds number (Williamson, 1996). At $x_1 = 0$ we prescribe an inflow velocity $u_1 = 1$, $u_2 = u_3 = 0$. With a viscosity of $\nu = 0.01$ the desired Reynolds number is obtained. At $x_1 = 10$ we prescribe a homogeneous Neumann outflow condition, at the cylinder boundary we impose a no-slip condition, while at the side walls we impose a free-slip condition. For the pressure we define homogeneous Neumann conditions at all boundaries. We discretize the domain with an unstructured tetrahedral mesh consisting of 38288 nodes, locally refined near the cylinder.

From the simulation we store $N = 400$ velocity snapshots, corresponding to all solution data available between $t_1 = 0$ and $t_N \approx 27.2$, which excludes the transient start-up time. On the left side of Figure 1 the field of the absolute velocity $|u| = (u_i u_i)^{\frac{1}{2}}$ at the final time t_N is shown.

We perform a POD of the selected snapshots using the method described above, with W equal to the identity matrix. The POD energies $\lambda_1, \ldots, \lambda_8$ are plotted semi-logarithmically in Figure 1 on the right.

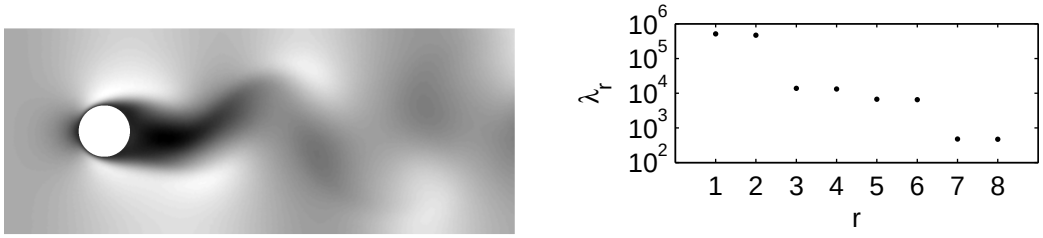

Figure 1: Test case with $Re = 100$. Left: Absolute velocity field at the final snapshot time. Right: POD energies $\lambda_1, \ldots, \lambda_8$ obtained using 400 snapshots.

Using the 8 most energetic modes and the snapshot mean, we create a POD-Galerkin model with constant model coefficients, according to (2). As an initial condition we take the projection of the first snapshot on the POD. We solve the model with a variable time step explicit Runge-Kutta method over the time interval in which we took the snapshots. For the comparison of the result with the snapshot data, we pick the mesh node next to the coordinates $(4, 2, 0.05)$, which is located downstream of the cylinder in the middle of the wake. In Figure 2 we plot the absolute velocity at this point together with the projection of the solution of the reduced model back to the physical space. It can be observed that the reduced model represents the dynamics of the full model well. The small visible differences could be explained by the numerical error present in the computation of the model coefficients as well as by neglecting boundary terms and less energetic modes.

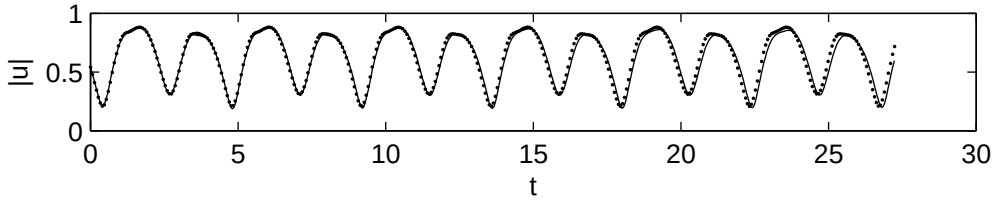

Figure 2: Absolute velocity computed with the POD-Galerkin model (solid line) compared to snapshot samples (dots) at the mesh node next to $(4, 2, 0.05)$ for the $Re = 100$ test case.

In Figure 3 we present the drag and lift coefficients obtained with the LES compared to the ones obtained with the reduced model. For the computation of these quantities the pressure field at the cylinder boundary is necessary. To compute the pressure field from the solution of the reduced model we employ the pressure model of Akhtar et al. (2009). We perform a snapshot POD of the pressure fluctuations and represent the pressure field at any time as a linear combination of the pressure snapshot mean and the pressure POD modes. To compute the pressure POD coefficients corresponding to the solution of the reduced model, we use a Galerkin projection of the pressure Poisson equation on the velocity and pressure modes. The pressure computations are carried out as a post-processing step after the time integration of the reduced model. Figure 3 shows that the result of the reduced model matches very well with the DNS in terms of drag and lift.

For this simple example of laminar vortex shedding, which is well approximated by a POD, the computing time for the reduced model is a very small fraction of the computing time of the DNS: Using a single 3 GHz core of a Sun X64 workstation, for the DNS 231 120 s of wall clock time were spent,

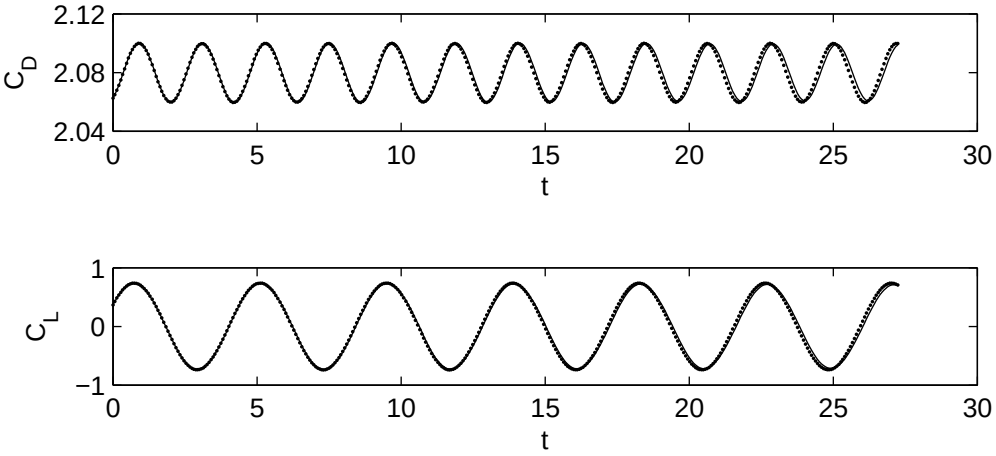

Figure 3: Drag coefficient $C_D(t)$ and lift coefficient $C_L(t)$ for the test case with $Re = 100$. The dots correspond to the values obtained from the LES, the solid lines correspond to the values obtained with the reduced model by employing the technique of Akhtar et al. (2009).

the computation of the velocity POD and the model coefficients took 28 s in total and the actual time integration took merely 5 s.

3.2 Turbulent flow

For the large-eddy simulation we use the following geometry: A cylinder of diameter $d = 1$ is centered at $x_1 = x_2 = 10$ aligned with the x_3-axis within a box-shaped domain of $0 \le x_1 \le 30$, $0 \le x_2 \le 20$ and $0 \le x_3 \le 3$. We prescribe the same boundary conditions as for the case with $Re = 100$, but set the viscosity to $\nu = 1/3900$ and choose $C_S = 0.15$. For the spatial discretization we use a mesh with 70731 nodes, locally refined at the cylinder and in the wake near the cylinder.

We take 400 snapshots between $t_1 = 0$ and $t_N \approx 28.6$ as input for the POD. A cutout of the absolute velocity field at t_N is shown in Figure 4 on the left. The POD energies of the first 64 modes are plotted on the right.

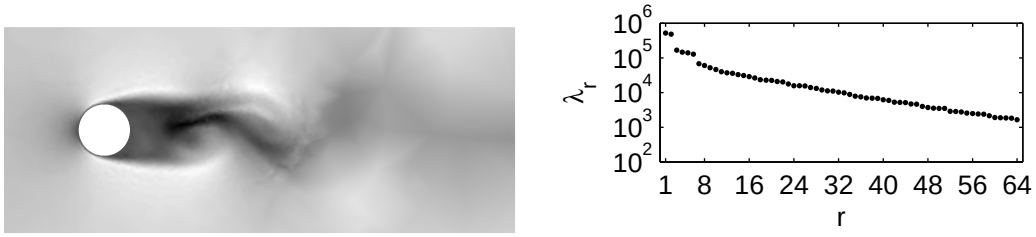

Figure 4: Test case with $Re = 3900$. Left: Cutout of the absolute velocity field at the final snapshot time. Right: POD energies $\lambda_1, \ldots, \lambda_{64}$ obtained using 400 snapshots.

We perform a run of the reduced model with Smagorinsky stabilization, as given by (4), using 64 POD modes and using the projection of the first snapshot on the POD as initial data. Employing a quadratic extrapolation of the time-dependent coefficients with an error tolerance of $\epsilon = 0.1$ we obtain solutions at 1090 time steps after performing 125 updates of C^{t}_{qr} and, incidentally, also 125 updates of D^{t}_{r}.

The turbulence naturally induces a high sensitivity of the solution with respect to the model parameters and the initial conditions, therefore we can expect a divergence between the solutions of the LES and of the reduced model. This is confirmed by the numerical solution, as shown in Figure 5, where the

absolute velocity at the mesh node next to $(12, 10, 0.15)$ is plotted for both cases. Despite the obvious differences, the extrema and the qualitative appearance of the plots are similar.

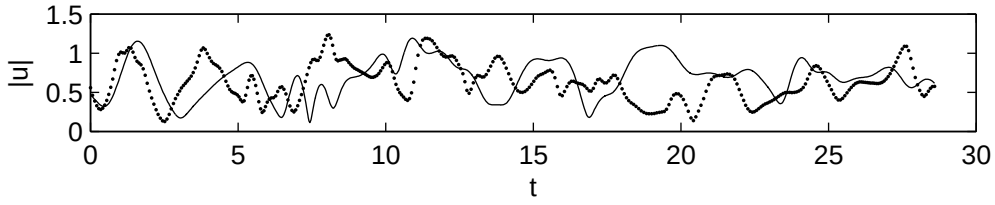

Figure 5: Absolute velocity computed with the updated POD-Galerkin model (solid line) compared to snapshot samples (dots) at the node next to (12,10,0.15).

For the presented test case, the reduced model is stable even for long integration times. In Figure 6 the result of the time integration over a time interval of 1000 time units is presented. As the solution leaves the time interval where the snapshots have been taken, an increase of the amplitude is visible. A reason for this may be that the POD using 400 snapshots is not fully converged, so more snapshots must be taken into account to get a more realistic long term behavior.

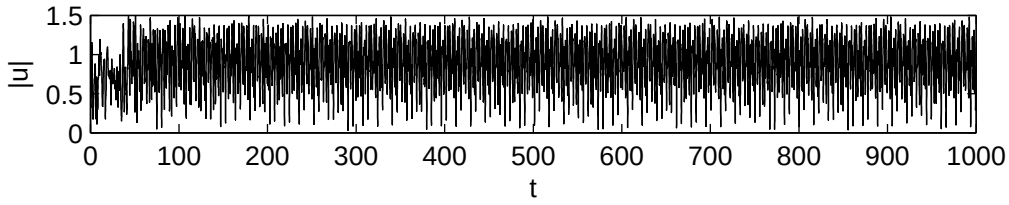

Figure 6: Result of the time integration of the updated reduced model at the same location as in Figure 5, but for a longer time interval. After the snapshot time interval the amplitude increases, but the long term behavior is stable.

The evolution of the drag and lift coefficients is shown in Figure 7. There is some phase error, but otherwise the results of the reduced model are qualitatively consistent with the results of the LES.

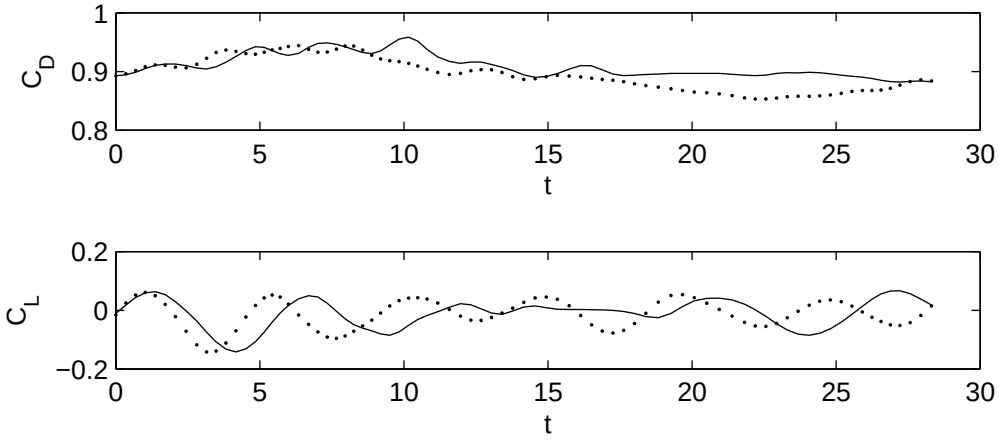

Figure 7: Drag and lift coefficients for the test case with $Re = 3900$. The dots correspond to the values of every fifth snapshot of the large-eddy simulation, the solid lines correspond to the reduced model.

Using the snapshots and the solution of the reduced model in the time interval where the snapshots have been taken, we compute the total kinetic energy spectral density of the velocity fluctuations. Figure 8 shows the average spectral density taken over all mesh nodes within the box $12 \leq x_1 \leq 13$,

$9.5 \leq x_2 \leq 10.5$ and $0.5 \leq x_3 \leq 2.5$. For low frequencies the spectra match very well. The peak of the energy spectrum indicates the Strouhal frequency. For the POD the peak is broader than for the LES and somewhat shifted towards lower frequencies. Above 4 times the Strouhal frequency the spectra diverge and the POD spectrum decreases sharply as a result of the truncation.

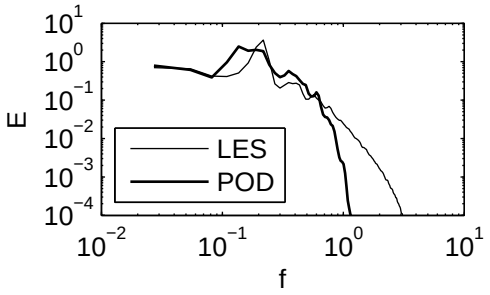

Figure 8: Total kinetic energy spectra of the velocity fluctuations in the cylinder wake for the test case with $Re = 3900$. In the low and medium frequency range the results of the updated POD-Galerkin model are comparable to the original LES; at higher frequencies the truncation of the POD leads to an energy decrease.

Finally, since a major purpose of the reduced modeling is the saving of computing time, we present some timings of the reduced LES model with updated coefficients. Using the same hardware as for the $Re = 100$ test case, the LES took 429 180 s in total, while building the reduced model took 921 s (including the computation of the POD, of the model coefficients and of the data necessary for the model updates) and the time integration took 4118 s of wall clock time. Because of the larger number of modes and of the additional work due to the model updates, the computing time is increased with respect to the test case with $Re = 100$. Still, the reduced model is much faster than the LES.

4 DISCUSSION

We have presented a reduced POD-Galerkin model of the Smagorinsky LES equations, taking the subgrid-scale model into account by updating the coefficients during the time integration. We have validated this model with a test case of the flow around a circular cylinder at $Re = 3900$. Although the solution of the reduced model does not exactly correspond to the original LES, the energy spectrum is reproduced well for low and medium frequencies and the drag and lift coefficients are captured reasonably. Even for long integration times, the numerical solution remains stable.

The results were obtained by using the same mesh-dependent filter width and the same Smagorinsky constant in the LES and in the reduced model. If there is a large gap between the LES filter width and the smallest scales present in the POD, then the damping introduced by updating the coefficients might be insufficient to stabilize the reduced model effectively. Therefore, the current model formulation requires the POD to be sufficiently fine or, conversely, the LES to be sufficiently rough in order to introduce enough dissipation. To make the model more robust, it might be possible to calibrate the Smagorinsky constant or the filter width with respect to the snapshots.

Using the example of the Smagorinsky model we have shown that, although computationally more demanding, it is possible to introduce an LES subgrid-scale model into a POD-Galerkin model. Further research has to reveal how other turbulence models perform in the context of reduced-order modeling and whether there are other ways to extend the standard Navier-Stokes based POD-Galerkin models using knowledge about the method with which the snapshots were obtained.

5 ACKNOWLEDGEMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under the grant SFB568/3.

References

I. Akhtar, A.H. Nayfeh, and C.J. Ribbens. On the stability and extension of reduced-order galerkin models in incompressible flows. *Theoretical and Computational Fluid Dynamics*, 23:213–237, 2009.

M. Bergmann, C.H. Bruneau, and A. Iollo. Enablers for robust POD models. *Journal of Computational Physics*, 228(2):516–538, 2009.

M. Buffoni, S. Camarri, A. Iollo, and M.V. Salvetti. Low-dimensional modelling of a confined three-dimensional wake flow. *Journal of Fluid Mechanics*, 569:141–150, 2006.

M. Couplet, C. Basdevant, and P. Sagaut. Calibrated reduced-order POD-Galerkin system for fluid flow modelling. *Journal of Computational Physics*, 207:192–220, 2005.

B. Erdmann, J. Lang, and R. Roitzsch. KARDOS user's guide. Tech. rep. ZR 02–42, Konrad-Zuse-Zentrum Berlin, 2002.

B. Galletti, C.H. Bruneau, L. Zanetti, and A. Iollo. Low-order modelling of laminar flow regimes past a confined square cylinder. *Journal of Fluid Mechanics*, 503:161–170, 2004.

B. Galletti, A. Bottaro, C.H. Bruneau, and A. Iollo. Accurate model reduction of transient and forced wakes. *European Journal of Mechanics – B/Fluids*, 26:354–366, 2007.

P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1996.

K. Kunisch and S. Volkwein. Control of the burgers equation by a reduced-order approach using proper orthogonal decomposition. *Journal of Optimization Theory and Applications*, 102:345–371, 1999.

J. Lang. Adaptive incompressible flow computations with linearly implicit time discretization and stabilized finite elements. In K.D. Papailiou, D. Tsahalis, J. Periaux, C. Hirsch, and M. Pandolfi, editors, *Computational Fluid Dynamics '98*, 1998.

B.R. Noack, P. Papas, and P. A. Monkewitz. The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows. *Journal of Fluid Mechanics*, 523:339–365, 2005.

L. Sirovich. Turbulence and the dynamics of coherent structures. parts I, II and III. *Quarterly of Applied Mathematics*, 45:561–571, 1987.

J. Smagorinsky. General circulation experiments with the primitive equations, I, The basic experiment,. *Monthly Weather Review*, 91:99–164, 1963.

H. Telib, M. Manhart, and A. Iollo. Analysis and low-order modeling of the inhomogeneous transitional flow inside a T-mixer. *Physics of Fluids*, 16:2717–2731, 2004.

C.H.K. Williamson. Vortex dynamics in the cylinder wake. *Annual Review of Fluid Mechanics*, 28:477–539, 1996.