Adaptively trained reduced-order model for acceleration of oscillatory flow simulations

O. F. Oxtoby

Advanced Computational Methods Group, Aeronautic Systems, Council for Scientific and Industrial Research, Pretoria, South Africa (ooxtoby@csir.co.za)

Abstract.

We present an adaptively trained Reduced-Order Model (ROM) to dramatically speed up flow simulations of an oscillatory nature. Such repetitive flowfields are frequently encountered in fluid-structure interaction modelling, aeroelastic flutter being one important application. The ROM is constructed using the method of snapshots and evaluated using both Proper Orthogonal Decomposition (POD) and the snapshots themselves as the basis modes. The incompressible Navier-Stokes equations are projected onto these basis modes using the method of Galerkin projection. While most ROM techniques try to speed up a sequence of similar simulations by first generating the ROM using selected representative runs, and then applying it to others, here it is generated on the fly in order to exploit the fact that individual simulations may themselves contain nearly-repetitive behaviour. In this work we propose a metric for determining when the ROM is accurate enough to use and when it needs to be augmented with further information from the full simulation. Thus, the process is fully automated and the amount of speed-up obtained depends on the degree to which the solution is repetitive in nature. The metric presented is a combination of monitoring of the overall residual as well as the mismatch between residuals of spatial and temporal terms generated by the ROM. Comparative accuracy and efficiency of flow simulations with and without the ROM are assessed.

Keywords: *Reduced-Order Model (ROM), Proper Orthogonal Decomposition (POD), Adaptive, Oscillatory flow.*

1. INTRODUCTION

The objective of this paper is to develop a reduced-order modelling technique to speed up the simulation of systems involving oscillatory or repetitive fluid flow, while maintaining acceptable engineering accuracy. A classic example of such a situation is in an aeroelastic flutter calculation where the vibration of aircraft wings and control surfaces in response to airflow must be characterised [1]. In order to account for the nonlinearities inherent in real airflows, as well as geometrical nonlinearities, a full computational fluid dynamics (CFD) simulation needs to be performed. However, since this is a fully time-dependent simulation it is very costly; in addition, it is usually necessary to perform many such simulations for

different conditions, and the resulting computational cost frequently makes it infeasible for use in practice [2], while the experimental cost is also often prohibitive [3].

While much work has been done in developing reduced-order models (ROMs) to characterise flutter response as a function of a parameter such as mach number or angle of attack [1,2], in this work we attempt to develop a method to speeding up each individual calculation by exploiting its repetitive or nearly-repetitive nature. This requires a ROM to automatically be trained as the simulation progresses, a topic on which there is little existing work. A technique for automatic adaptation of a ROM for mesh-movement was presented by Bogaers *et al.* [4] and shown to be very effective. Additional complexity is introduced, however, when solving time-dependent equations such as the Navier-Stokes equations as inaccuracy is accumulated as the solution progresses, and care must be taken to ensure that, when snapshots are added due to an inadequate ROM solution, they are not contaminated by that accumulated error. Accordingly, in the procedure described below the aim is to monitor the level of accuracy and ensure that it stays within nominated bounds. A similar technique applied to time-dependent Fisher and Ginzburg-Landau equations has been presented by Rapún and Vega [5], but not to the author's knowledge for the Navier-Stokes equations.

The equations being solved here are the viscous, incompressible Navier-Stokes equations, i.e.

$$\rho \frac{\partial u_i}{\partial t} = -\rho \frac{\partial}{\partial x_j} u_i u_j - \frac{\partial p}{\partial x_i} + \mu \frac{\partial^2 u_i}{\partial x_j^2} \tag{1}$$

and

$$\frac{\partial u_j}{\partial x_j}. \tag{2}$$

The structure of this paper is as follows. In Section 2, the theory behind the generation of ROMs via the method of snapshots with and without POD is presented. Next, in Section 3 the novel techniques for enhancing the ROM on-the-fly with new detailed calculations are presented. Numerical results are then presented in Section 4 and finally, we conclude with Section 5.

2. REDUCED-ORDER MODEL FOR FLUID FLOW

Firstly, the reduced-order model is built using Proper Orthogonal Decomposition and the Method of Snapshots. Both are standard techniques and are well described in many reference, see e.g. [6]. In short the techniques involve taking snapshots of the flow fields and performing a principal-components analysis on them to make them orthogonal and to capture the maximum variation in as few modes as possible. The only thing to note is that the ROM is built based on the difference between the snapshots and their average value; this was found to generate a more accurate solution as the numerical dissipation was not able to diminish the energy towards zero, but instead ensures that the correct boundary conditions are always maintained.

Secondly, we consider a ROM built using the snapshots themselves as the basis modes, i.e. without generating principal orthogonal modes using POD. Although this technique no longer captures the maximum variation using the fewest modes, the advantage is that it allows us to dramatically reduce the time taken to build the ROM. Due to the adaptive nature of the

ROM considered here, the continually repeated rebuilding process has to be fast in order to realise overall speed gains, as will be shown.

2.1. Galerkin projection

In this subsection we present the projection of the original, continuous, governing equations onto the basis modes, when in actual fact we are dealing with the discrete numerical realisation of these equations. This is important for consistency since in the adaptive code, one is constantly switching back and forth between the detailed numerical solution and the ROM. However, presenting the governing equations only is done for clarity of notation, with special considerations for the numerical equivalents being noted where applicable.

2.1.1 Using POD modes

We begin by writing the velocity and pressure as the average of snapshots plus a linear combination of the POD modes:

$$\begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \begin{pmatrix} \overline{\mathbf{u}} \\ \overline{p} \end{pmatrix} + \sum_{l=1}^{L} \gamma_l \begin{pmatrix} \boldsymbol{\psi}^l \\ \phi^l \end{pmatrix}. \tag{3}$$

Here, γ_l are the coefficients in the ROM, while $\boldsymbol{\psi}_l$ and ϕ_l are the basis modes for the ROM. L is the number of retained modes in the POD. Using the method of Galerkin Projections (see e.g. [6]), we substitute these expressions into the Navier-Stokes equations and insist that the projection onto each of the basis modes is zero; i.e. that that governing equations are satisfied on the subspace spanned by the basis modes. Substituting (3) into the continuity equation, we obtain

$$\boldsymbol{\nabla} \cdot \left(\overline{\mathbf{u}} + \sum_{l=1}^{L} \gamma_l \boldsymbol{\psi}_l \right) = 0. \tag{4}$$

However, because the average velocity as well as every basis mode is a linear combination of the original snapshots, all of which are divergence-free, the equation above is identically satisfied. Note that, in the numerical realisation of the continuity equation the velocity is divergence-free only to truncation accuracy rather than convergence accuracy. It is only the forward-projected velocities which are truly divergence-free, and they contain contributions from the pressure due to the pressure-projection method used. We refer to these divergence-free fluxes as the 'face fluxes'. Therefore, as long as the pressure and velocity snapshots are kept in the same proportion, the discrete version of the continuity equation is also identically satisfied by the expansion (3). This is the reason why a single expansion coefficient has to be used for velocity and pressure in Eq. (3).

Now performing the Galerkin projection on the momentum equations, we obtain

$$
\rho\dot{\gamma}_k(\psi_i^k, \psi_i^k) = -\rho\left(\frac{\partial}{\partial x_j}\overline{u}_i\overline{u}_j, \psi_i^k\right) - \rho\sum_{l=1}^{L}\gamma_l\left(\frac{\partial}{\partial x_j}\overline{u}_i\psi_j^l, \psi_i^k\right) - \rho\sum_{l=1}^{L}\gamma_l\left(\frac{\partial}{\partial x_j}\overline{u}_j\psi_i^l, \psi_i^k\right)
$$

$$
-\rho\sum_{l=1}^{L}\sum_{m=1}^{L}\gamma_l\gamma_m\left(\psi_i^l\psi_j^m, \psi_i^k\right) - \left(\frac{\partial\overline{p}}{\partial x_i}, \psi_i^k\right) - \sum_{l=1}^{L}\gamma_l\left(\frac{\partial}{\partial x_i}\phi^l, \psi_i^k\right)
$$

$$
+ \left(\frac{\partial^2}{\partial x_j^2}\mu\overline{u}_i, \psi_i^k\right) + \sum_{l=1}^{L}\gamma_l\left(\frac{\partial^2}{\partial x_j^2}\mu\psi_i^l, \psi_i^k\right) \quad (5)
$$

where (a_i, b_i) is shorthand for $\int_V \mathbf{a}\cdot\mathbf{b}dV$ and V is the solution domain. The pressure terms (last two terms on the second line) are often ignored (thereby eliminating pressure as a variable) because they can be reduced to boundary integrals if the velocity snapshots ψ^k are divergence-free. However, in our case it is only the face flux and not the velocities which are truly divergence free, and in any case the boundary integral is only zero if the pressure at all boundaries is the same. Regardless, for our application it would be inconvenient to have to reconstruct pressure from the momentum equation whenever we wished to switch from the ROM back to the detailed solution, so we therefore make no effort to eliminate it from the model.

2.1.2 Using snapshots as basis modes

When using the snapshots themselves as basis modes, we do not consider deviations from the average as we wish to minimise re-computation by being able to add extra modes without affecting the existing ones. We therefore expand the primary variables as

$$
\begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \sum_{m=1}^{M}\gamma_m\begin{pmatrix} \boldsymbol{u}^m \\ p^m \end{pmatrix}, \quad (6)
$$

where γ_l again denote the expansion coefficients, $\boldsymbol{u}_l$ and p_l the snapshots, and M is the number of snapshots. Again using the method of Galerkin Projections, the continuity equation is identically satisfied, while the only difference in the momentum equations is the presence of off-diagonal terms on the left-hand side:

$$
\rho\sum_{m=1}^{M}\dot{\gamma}_m(u_i^m, u_i^k) = -\rho\sum_{l=1}^{M}\sum_{m=1}^{M}\gamma_l\gamma_m\left(u_i^l u_j^m, u_i^k\right)
$$

$$
-\sum_{m=1}^{M}\gamma_m\left(\frac{\partial}{\partial x_i}p^m, u_i^k\right) + \sum_{m=1}^{M}\gamma_m\left(\frac{\partial^2}{\partial x_j^2}\mu u_i^m, u_i^k\right). \quad (7)
$$

Solution of the above system of equations requires inversion of the matrix on the left-hand side, but no additional complications arise from the non-orthogonality of the modes.

The virtue of the above formulation is that, not only does it eliminate the need to perform a POD when a snapshot is added, but only a small fraction of the coefficients in the above system of equations [i.e. the bracketed terms in Eq. (7)] need to be recomputed when the model is rebuilt [$\mathcal{O}(M^2)$ coeffients rather than $\mathcal{O}(L^3)$ in the POD case].

So that (as described above) the coefficients in Eq. (7) do not all have to be recomputed whenever a new snapshot is added, the snapshots have not been separated from the average (as in the POD case). However, it is still desirable for the expansion (6) to represent deviations from the average solution rather than from zero, for the reasons mentioned before. To achieve this in effect, after every time-step of Eq. (7) we shift the expansion coefficients γ_l by a constant offset in order to satisfy

$$\sum_{l=1}^{M} \gamma_l = 1. \tag{8}$$

It can be shown that this procedure is equivalent to minimising the residual of Eq. (1) with respect to the variables γ_l, subject to the constraint (8).

3. AUTOMATIC ADAPTATION

The procedure adoped to automatically update the reduced-order model from full finite-volume calculations is as follows.

1. Solve the equations in full detail for a nominated time known as the snapshot interval ΔT.

2. Add the current velocity and pressure fields as snapshots and regenerate the ROM.

3. Solve using the the ROM, evaluating the accuracy every timestep (Δt). If accuracy is below a set threshold, go back to the beginning of the timestep and continue from step 1.

The aim is to solve the system using the ROM while it is adequate, but to detect the point at which the true solution diverges from the subspace of the ROM. Once this happens, the system is solved in full detail for and snapshot interval ΔT after which a new snapshot is added to the ROM and it is attempted to continue solving with it. The snapshot interval should be on the scale of the physical timescales of the problem or larger, so that the detailed solution of the system has a chance to deviate meaningfully from the too-restrictive ROM subspace from which it was re-started. On the other hand, making it too long will waste unneccesary computation time on the detailed solution.

The challenge is to identify an accurate heuristic to detect when the detailed solution 'wants to' diverge from the subspace of the ROM, without actually calculating it. The most obvious approach is to detect how accurately the ROM is approximating the full solution by monitoring the full residual (L_2-norm). However, experience shows that this is a fairly poor predictor, unable to differentiate between a plausible (albeit approximate) solution and one which is completely physically unrealistic.

A much more useful metric is obtained by comparing the left- and right-hand sides of the momentum equation when the ROM solution is substituted in. The two are identical when projected onto the POD modes, of course, but comparing them when the reconstructed fields are substituted in gives a comparison of where the full solution 'wants to go' (RHS) versus where it was allowed to go within the ROM subspace (LHS). The metric used is

$$\Delta = \frac{\int_V |RHS| dV}{\int_V |\frac{d\mathbf{u}}{dt}| dV}. \tag{9}$$

A value of $\Delta > 1$ indicates that the ROM subspace is constraining the solution to some degree.

To maintain the accuracy of the solution, we monitor both metrics – the residual and Δ – maintaining them both below preselected limits.
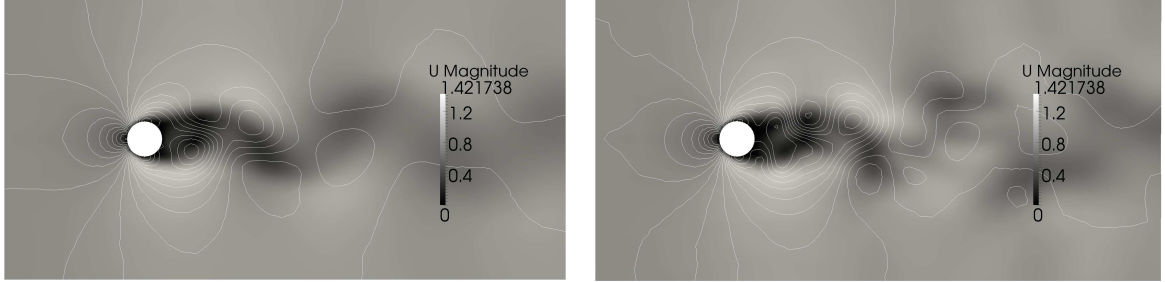
4. RESULTS

Figure 1. Velocity magnitude contours (shading) and pressure contour (lines spaced at 0.06 Pa intervals between -0.6 Pa and 0.6 Pa), comparing the full finite-volume solution (left) with the adaptive ROM solution (right) at the end of the simulation (75 s).

The test-case we consider consists of the periodic shedding of vortices from a cylinder as depicted in Fig. 1. The mesh consists of 98 000 cells. A constant velocity of 1 ms^{-1} is imposed one the left hand boundary, while the right hand boundary is at a constant pressure of zero. The lateral boundaries are slip boundaries and the cylinder wall has a viscous (no-slip) boundary condition.

This test case was run initially with the standard finite-volume incompressible solver `icoFoam` in OpenFOAM, and then using the adaptive ROM algorithm. The vortex-shedding initially builds up and then settles into a periodic regime, providing a good test of the adaptive capabilities of the ROM. For this problem the convergence metrics were chosen as $\Delta < 1.25$ and Residual < 0.08. For the POD ROM, the number of POD modes selected was 16.

Figure 2 compares the solutions obtained with the full simulation to those obtained using the adaptive ROM with snapshots used as the basis modes. First, in the top panel, the solutions themselves are compared. Whilst there is a considerable difference in the initial transition to limit-cycle oscillation, this is to be expected as the problem starts off in a state of unstable equilibrium. The rate at which it leaves this state is dependent initially on the numerical noise present and so will be affected by the different numerical schemes used. The steady-state oscillation regime is comparable to engineering accuracy. The ROM here consists of only 42 degrees of freedom compared to the thousands used in the detailed solution.

In the middle panel of Fig. 2, computational time is compared as a function of simulation time, and compared with that for the ROM with orthogonal POD modes. The long, almost-horizontal lines show how quickly the ROM runs once it has been generated. The speedup in this region is 56 times compared to the full solution. This, of course, includes the time taken to monitor the full residual in order to calculate the accuracy heuristics. However, for the POD ROM the steep staircase in the dotted line prior to 36 s can be seen to be the
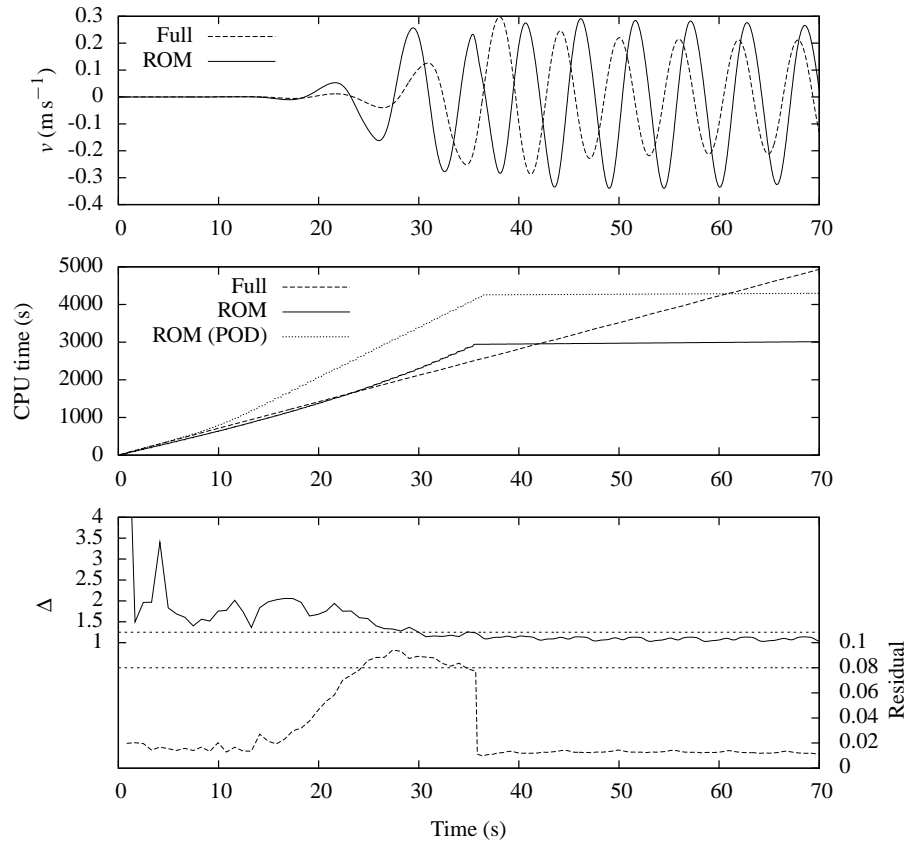
Figure 2. Results of the vortex-shedding cylinder test-case. In the top panel, the vertical velocity at centre of the domain outflow boundary is shown for the full finite-volume solution compared to the ROM (using snapshots as non-orthogonal basis modes). Comparative CPU time taken to solve is shown as a function of simulation time in the middle panel, both for POD and non-orthogonal basis modes. The lower panel depicts the values of the convergence metrics Δ and Residual as a function of time, again for the non-orthogonal case. The convergence thresholds selected for this run are denoted by the dotted lines.

limiting factor in the speed of the solution, and is a consequence of the time taken to regenerate the ROM whenever a snapshot is added – most of which is taken up in regenerating Galerkin projection coefficients. The result is only a 23% speedup overall. The ROM with non-orthogonal modes, by contrast, has visibly less computational overhead in the 'learning' phase and yields an overall speedup in computational time of 67% for this problem.

The lower panel of Fig. 2 shows the convergence metrics, Δ and Residual as a function of time along with the thresholds selected for this problem. Note that initially, the Δ metric large but the residual metric is nonetheless within the required tolerance; this means that the ROM solution solves the equations with acceptable accuracy but on too restricted a subspace. Later the Δ metric falls below the designated tolerance but the residual rises, implying that the subspace spanned by the basis modes is large enough to encompass the required time dynamics but not in sufficient detail.

5. CONCLUSIONS AND FUTURE WORK

In this work we have proposed a new scheme for generating reduced-order models on the fly, and developed a heuristic for determining when the basis modes need to be augmented with further snapshots of the detailed solution.

As mentioned, the regeneration of the ROM and more particularly the calculation of Galerkin coefficients is a major bottleneck in the speed of the ROM solution. To alleviate this it was shown to be worthwhile to relax the condition that the basis modes should be orthogonal. This makes the ROM equation (5), somewhat more costly to solve, however, this is vastly outweighed by the advantage that all but one of the basis modes stay the same whenever the model is augmented, drastically reducing the number of new Galerkin coefficients to be calculated. Despite this, when the number of snapshots becomes very large, the number of Galerkin coefficients to be calculated can still become prohibitive. Therefore it is proposed that a limit be placed on the number of snapshots stored, with a 'scoring' system used to determine which to discard based on which has played the smallest part in the reconstructed solution.

One drawback of the current system is that it is not obvious how to select the residual threshold to provide a good tradeoff between accuracy and computation time. This question will be considered in future work.

6. REFERENCES

[1]   Lieu T., Farhat C., Lesionne M., "Reduced-order fluid/structure modeling of a complete aircraft configuration". *Comp. Meth. Appl. Mech. Eng.* 195, 5730-5742, 2006.

[2]   Farhat C., Geuzaine P., Brown G., "Application of a three-field non-linear fluid–structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter", *Comput. Fluids* 32, 329, 2003.

[3]   Epureanu B.I., "A parametric analysis of reduced order models of viscous flows in turbo-machinery", *J. Fluids Struct.* 17 971982, 2003.

[4]   Bogaers A.E.J., Kok S, Malan A.G., "Highly efficient optimization mesh movement method based on proper orthogonal decomposition", *Int. J. Numer. Meth. Engng* 86 935-952, 2011.

[5]   Rapún M.-L., Vega J.M., "Reduced order models based on local POD plus Galerkin projection", *J. Comp. Phys.* 229 3046-3063, 2010.

[6]   Bogaers A.E.J., "Reduced order modeling techniques for mesh movement strategies as applied to fluid structure interactions", Masters thesis, University of Pretoria, 2010.