8th GRACM International Congress on Computational Mechanics,
Volos, 12 July - 15 July 2015

LOW-COST, UNSTEADY CONTINUOUS ADJOINT METHOD FOR OPTIMAL FLOW CONTROL USING JETS

Ioannis S. Kavvadias[1], Christos K. Vezyris[1], Evangelos M. Papoutsis-Kiachagias[1] and Kyriakos C. Giannakoglou[1]

[1]Parallel CFD & Optimization Unit, School of Mechanical Engineering,
National Technical University of Athens,
Athens, Greece
e-mail: kavvadiasj@hotmail.com, cvezyris@gmail.com, vaggelisp@gmail.com, kgianna@central.ntua.gr
web page: http://velos0.ltt.mech.ntua.gr/research

Keywords: Unsteady continuous adjoint, Approximate models, Incremental SVD, Flow control, Optimization.

Abstract:

The paper evaluates approximation models for the unsteady flow fields reconstruction, needed for the solution of the unsteady adjoint equations, in the context of optimization problems in fluid mechanics. When full storage of the computed flow fields at all time-steps cannot be afforded, it is a common practice to resort to the check-pointing technique. This consists of selectively saving a small number of flow field snapshots, to be used as the starting points for the re-computation of any other instantaneous flow field required during the backward in time integration of the adjoint equations. Check-pointing, though accurate, is a trade-off between computational cost and storage. Alternatively, while solving the adjoint equations marching backwards in time, flow variables can be approximated at each grid cell center, instead of re-computing their exact values. Three approximation models are considered, based on cubic B-splines, Fourier series and the incremental Singular Value Decomposition. In this paper, the unsteady continuous adjoint method is used for the jet-based flow control optimization in external aerodynamics, though the described models could be applied in shape optimization as well. Here, a number of pulsating jets arranged on the body contour contribute to the reduction in the time-averaged drag induced by the generated Karman vortices. The amplitudes of the pulsating jets, having fixed positions, phases and frequencies, are considered as optimization variables. The optimal solutions computed by using the approximation models are compared to those found by the check-pointing technique (considered as reference), in terms of accuracy and overall computational cost.

1 INTRODUCTION

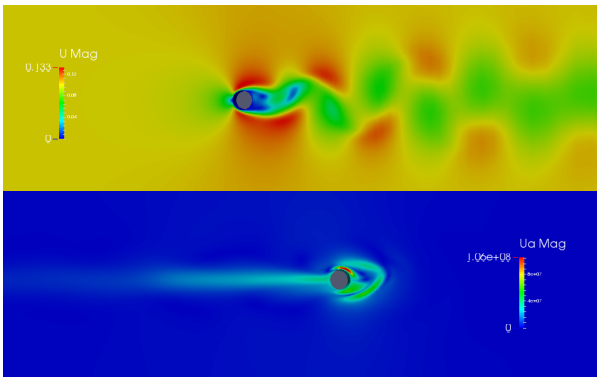
Given an optimization problem in fluid mechanics, expressed by the objective function to be minimized, the PDEs of the flow model and a set of design variables, the most efficient way to compute the gradient of the objective function with respect to (w.r.t.) the design variables is the adjoint method [9, 5]. In the continuous adjoint method [7, 8], the adjoint PDEs are firstly derived and then discretized. For the problems under consideration, the unsteady Navier-Stokes (N-S) equations for the laminar flow of an incompressible fluid are used; the objective function is a time-averaged performance metric, such as the drag. Though only laminar flows are considered, work conducted by the same group guarantees that the continuous adjoint method can include exactly differentiated turbulence models [8].

In unsteady flows, the adjoint information travels backwards in time w.r.t. the primal (flow) one [6]. The numerical solution of the unsteady adjoint equations requires the availability of the instantaneous flow fields. The off-the-shelf solution to this problem would be to have the entire time series of the flow fields stored. However, especially in real-world large-scale optimization problems, this cannot be a viable solution due to memory limitations.

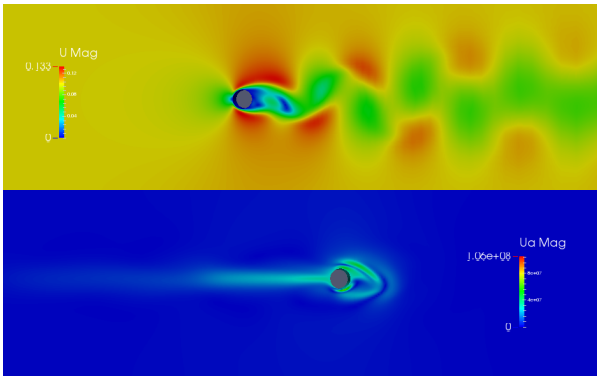
As a remedy to this problem, check-pointing techniques [3, 11] are widely used; they reduce memory consumption, by increasing though the computational cost due to the necessary re-computations, which all start from the closest

stored check-point. Even though binomial check-pointing has minimal cost for a given memory allocation, it may still be too expensive. In order to overcome limitations regarding both memory and computational cost, alternatives based on approximations to the time-varying flow field can be devised. These approximation models could include simple linear models or others of higher accuracy. In this paper, models based on cubic B-splines, Fourier series and the incremental Singular Value Decomposition (iSVD) are used and compared on the same cases.

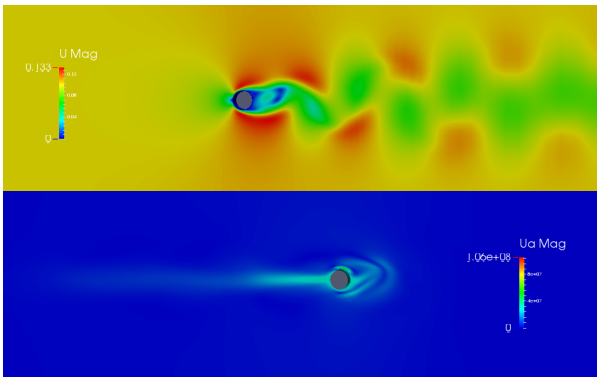
This paper begins by presenting the development of the unsteady continuous adjoint equations, boundary conditions and sensitivity derivatives. Then, the models used to approximate the unsteady flow fields are presented, followed by a discussion on their performances in the considered cases.



(a)

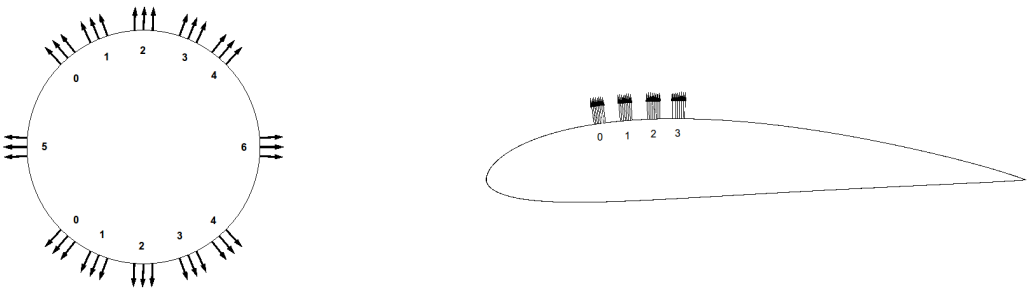


(b)



(c)

Figure 1: Flow around a cylinder; uncontrolled case. Three equidistant snapshots of the flow velocity (top half in each snapshot) and the corresponding adjoint velocity magnitude (bottom half in each snaphot) computed over a period of $T \approx 0.59s$ (natural period of the phenomenon). These snapshots were taken after a periodic flow was established. The generation of Karman vortices is shown.



(a) (b)

Figure 2: Fixed jet locations, (a) along the surface of the cylinder and (b) along the surface of the NACA4415 airfoil. Five pairs of jets (0-4) and two single jets (5 and 6) in the first case and four single jets in the second one are considered.

2 FLOW MODEL AND OBJECTIVE FUNCTIONS

The flow is modeled by the unsteady laminar N-S equations for an incompressible fluid. The primal equations read

$$\begin{aligned} R_i^v &= \frac{\partial v_i}{\partial t} + v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + \frac{\partial p}{\partial x_i} = 0\,, \quad i = 1, 2(, 3) \\ R^p &= -\frac{\partial v_j}{\partial x_j} = 0 \end{aligned} \tag{1}$$

where v_i and p stand for the Cartesian velocity components and the static pressure divided by the density, respectively. Eqs. (1) are solved by the PISO algorithm [4] in OpenFOAM©; a collocated, cell-centered finite-volume discretization scheme on unstructured meshes is implemented.

The problem considered in the this paper is the flow control optimization by means of pulsating jets, where the objective function is the time-averaged squared drag exerted on the body [2, 12]. The Cartesian velocity components of each jet are given by

$$v_\lambda^n = \left(A^n \sin\left(2\pi f^n \left(t - f_0^n\right)\right) - A^n\right) n_\lambda\,, \quad \lambda = 1, 2 \tag{2}$$

where n is the jet counter, A^n the amplitude, f^n the frequency and f_0^n the phase of each jet. All jets are constantly aligned with the normal to the wall unit vector n_λ as seen in fig. 2. Positive A^n corresponds to blowing (since n_λ points outwards) and negative A^n to suction. In the studies presented here, only the jet amplitudes, A^n, are considered as design variables, keeping constant the frequencies and phases $f^n = \frac{v_\infty}{d}$ and $f_0^n = 0$, as in [6], v_∞ is the infinite flow velocity and d is an appropriate length (e.g. the diameter of the cylinder).

The time-averaged (squared) drag force, to be minimized, is

$$J = \frac{1}{2T}\int_T D^2(t)dt \tag{3}$$

where T is the flow period. In the uncontrolled case, T equals the Karman vortices period. In the controlled cases, T becomes equal to $0.1s$. $D(t)$ is the instantaneous drag force

$$D(t) = \int_{S_w}\left[pn_i - \nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) n_j - |v_j n_j|\, v_i\right] r_i dS \tag{4}$$

where r_i are the components of the unit vector aligned with the infinite flow velocity and S_w is the solid wall boundary. The last term in eq. (4) stands for the jets contribution to the force exerted to the body.

The derivative of the 'mean drag' objective function w.r.t. $b_n \equiv A^n$ is

$$\begin{aligned} \frac{\delta J}{\delta b_n} &= \frac{1}{T}\int_T\int_{S_w} D\Bigg(-\frac{\partial}{\partial b_n}\left[\nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] n_j \\ &+ \frac{\partial v_i}{\partial b_n}|v_j n_j| + \frac{v_k n_k}{|v_\lambda n_\lambda|}\frac{\partial v_j}{\partial b_n} n_j v_i + \frac{\partial p}{\partial b_n} n_i\Bigg) r_i dS dt \end{aligned} \tag{5}$$

3 THE UNSTEADY CONTINUOUS ADJOINT METHOD

The derivation of the unsteady continuous adjoint equations starts with the augmented objective function L which reads

$$L = J + \int_T \int_\Omega u_i R_i^v \, d\Omega dt + \int_T \int_\Omega q R^p \, d\Omega dt \tag{6}$$

where u_i and q are the adjoint velocities and adjoint pressure, respectively. Differentiating L w.r.t. b_n and applying the Leibniz theorem, gives

$$\frac{\delta L}{\delta b_n} = \frac{\delta J}{\delta b_n} + \int_T \int_\Omega u_i \frac{\partial R_i^v}{\partial b_n} \, d\Omega dt + \int_T \int_\Omega q \frac{\partial R^p}{\partial b_n} \, d\Omega dt + \int_T \int_S (q R^p + u_i R_i^v) \frac{\delta x_k}{\delta b_n} n_k \, dS dt \tag{7}$$

Since $\frac{\delta x_k}{\delta b_n} = 0$, the last integral in eq. (7) vanishes. From eq. (7) the field adjoint equations are derived. By satisfying them, the field integrals depending on variations in the flow variables w.r.t. b_n are eliminated. The field adjoint equations are

$$R_i^u = -\frac{\partial u_i}{\partial t} - v_j \frac{\partial u_i}{\partial x_j} + u_j \frac{\partial v_j}{\partial x_i} + \frac{\partial q}{\partial x_i} - \frac{\partial}{\partial x_j} \left[\nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] = 0 \tag{8}$$

$$R^q = -\frac{\partial u_i}{\partial x_i} = 0 \tag{9}$$

After satisfying the field adjoint equations, eq. (7) becomes

$$\begin{aligned} \frac{\delta L}{\delta b_n} &= \frac{\delta J}{\delta b_n} + \int_\Omega \left[v_i \frac{\partial v_i}{\partial b_n} \right]_0^T d\Omega + \int_T \int_S \left[u_i v_j n_j + \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j - q n_i \right] \frac{\partial v_i}{\partial b_n} dS dt \\ &+ \int_T \int_S u_j n_j \frac{\partial p}{\partial b_n} dS dt - \int_T \int_S \nu u_i \frac{\partial}{\partial b_n} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) n_j dS dt \end{aligned} \tag{10}$$

where, $S = S_\infty \cup S_w$ (S_∞ is the freestream boundary) is the domain boundary. The adjoint boundary conditions result by substituting eq. (5) into eq. (10) and eliminating of the boundary integrals that include variations in the flow variables w.r.t. b_n. The instantaneous adjoint boundary conditions are $u_i = -\frac{D(t)}{T} r_i$ (along S_w) and $u_i = 0$ (along S_∞). Over the whole domain Ω, the initial adjoint conditions are $u_i|_{t=T} = 0$.

The remaining terms give the sensitivities of J w.r.t. the control variables $b_n \equiv A^n$ which are

$$\begin{aligned} \frac{\delta J}{\delta b_n} &= \int_T \int_{S_w} \left[u_i v_j n_j - u_i |v_j n_j| + \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) n_j \right. \\ &- \left. q n_i - \frac{v_k n_k}{|v_\lambda n_\lambda|} u_j v_j n_i \right] (sin(2\pi f^n (t - f_0^n)) - 1) n_i dS dt \end{aligned} \tag{11}$$

4 APPROXIMATION MODELS FOR THE FLOW TIME-SERIES

According to eqs. (9) – (11), the numerical solution of the field adjoint equations and the computation of the sensitivity derivatives require the velocity and pressure fields at each time instant. The straightforward way to do so is by storing all the instantaneous flow fields. However, the 'full storage' approach may become highly memory demanding.

Nowadays, check-pointing techniques [3, 11] are the most common alternative. In these techniques, some instantaneous flow fields are selectively stored during the forward sweep (i.e. the numerical solution of the N-S equations). Whenever a non-stored flow field is needed, the N-S equations are solved anew, starting from the last available check-point and the fields are re-computed. Binomial check-pointing guarantees minimum re-computations due to the optimal distribution of snapshots.

Apart from full storage and check-pointing techniques, approximation models can be utilized to approximate (instead of re-computing) the flow fields at each time step during the backward integration in time of the adjoint equations. In this paper, three approximation models are implemented. The first one relies upon cubic B-splines.

Given a series of stored snapshots at time instant t_i, $i = 0 \dots m$, in the adjoint loop, unfolding backwards in time, velocity and pressure fields (ω) are approximated by a piecewise polynomial, such as

$$\omega_t = \omega_i + \left[\omega_{i+1} - \omega_i - \frac{1}{6}M_{i+1} - \frac{1}{3}M_i\right]\tau + \frac{1}{2}M_i\tau^2 + \frac{1}{6}\left(M_{i+1} - M_i\right)\tau^3 \tag{12}$$

where the coefficients M_i are computed by solving a tri-diagonal system and $\tau = \frac{t-t_i}{t_{i+1}-t_i}$.

The second model relies upon Fourier series to approximate the time series of the unsteady flow fields. The number of harmonics, h_{max}, is user-defined. This can be expressed as

$$\omega_t = \frac{1}{2}a_0 + \sum_{h=1}^{h_{max}} a_h cos\left(ht\right) + \sum_{h=1}^{h_{max}} b_h sin\left(ht\right) \tag{13}$$

where, $a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} \omega(t)dt$, $a_h = \frac{1}{\pi}\int_{-\pi}^{\pi} \omega(t)cos\left(ht\right)dt$, $b_h = \frac{1}{\pi}\int_{-\pi}^{\pi} \omega(t)sin\left(ht\right)dt$.

The third model is based on the incremental Singular Value Decomposition (iSVD). The interested reader is referred to [10] for more information on the specifics of the iSVD model. Below, only the basics of this method are presented. For the snapshot matrix $A \in \mathbb{R}^{c\times m}$ (c is the total number of the grid cells and m the number of snapshots), SVD is performed by decomposing A as $A = U\Sigma V^T$, where $U \in \mathbb{R}^{c\times c}$ and $V \in \mathbb{R}^{m\times m}$ are orthogonal matrices and $\Sigma \in \mathbb{R}^{c\times m}$ is a diagonal matrix formed by the singular values of A in descending order ($\sigma_1 \geq \sigma_2 \geq \dots \sigma_{min(c,m)} \geq 0$). Storage requirements can be reduced by considering a truncation point k. In such a case, the decomposition of the snapshot matrix A yields

$$A = U_{c\times k}\Sigma_{k\times k}V^T_{k\times m} \tag{14}$$

where $k \ll m$ is user-defined. Unfortunately, (standard) SVD requires the complete snapshot matrix A to be available. In unsteady problems, this would result practically to full storage, without any apparent advantage. The incremental update (at each time step) of the matrices generated by the SVD [10] is proposed instead. The purpose of the incremental approach [1] is to process the information for the velocity and pressure fields generated at each time step of the forward sweep, by re-adjusting accordingly the U, V, Σ matrices, with the same maximum rank, k.

Due to the decreasing order of the singular values in the Σ matrix, the last entry is eliminated when the solution at the next time step becomes available. Then, the SVD matrices are updated according to the following algorithm [10]: for every new vector, $\omega \in \mathbb{R}^c$, formed by the flow variables at the new instant, we compute

$$p' = U^T\omega \qquad\qquad r' = \omega - U\omega \tag{15}$$

Then, the previous decomposition $U\Sigma V^T$ is updated by appending the new vector ω

$$\left[U\Sigma V^T \;\; \omega\right] = \left[U \;\; \frac{r'}{\|r'\|}\right]\begin{bmatrix} \Sigma & p' \\ 0 & \|r'\| \end{bmatrix}\begin{bmatrix} V & 0 \\ 0 & 1 \end{bmatrix}^T \tag{16}$$

where

$$\begin{bmatrix} \Sigma & p' \\ 0 & \|r'\| \end{bmatrix} = U'\Sigma'V'^T \tag{17}$$

leading to the updated U, Σ and V matrices

$$U^{new}_{c\times k} = \left[U \;\; \frac{r'}{\|r'\|}\right]U' \qquad\qquad \Sigma^{new}_{k\times k} = \Sigma' \qquad\qquad V^{new}_{m\times k} = \begin{bmatrix} V & 0 \\ 0 & 1 \end{bmatrix}V' \tag{18}$$

5 APPLICATIONS–COMPARISONS OF THE APPROXIMATION MODELS

The approximation models presented above are used to support the adjoint-based solution of two optimization problems. The first one is about the flow control optimization of the well-documented periodic flow around a cylinder with Re=100, shown in fig. 1. The 2-D computational grid is structured-like with $\sim 10^4$ quadrilateral

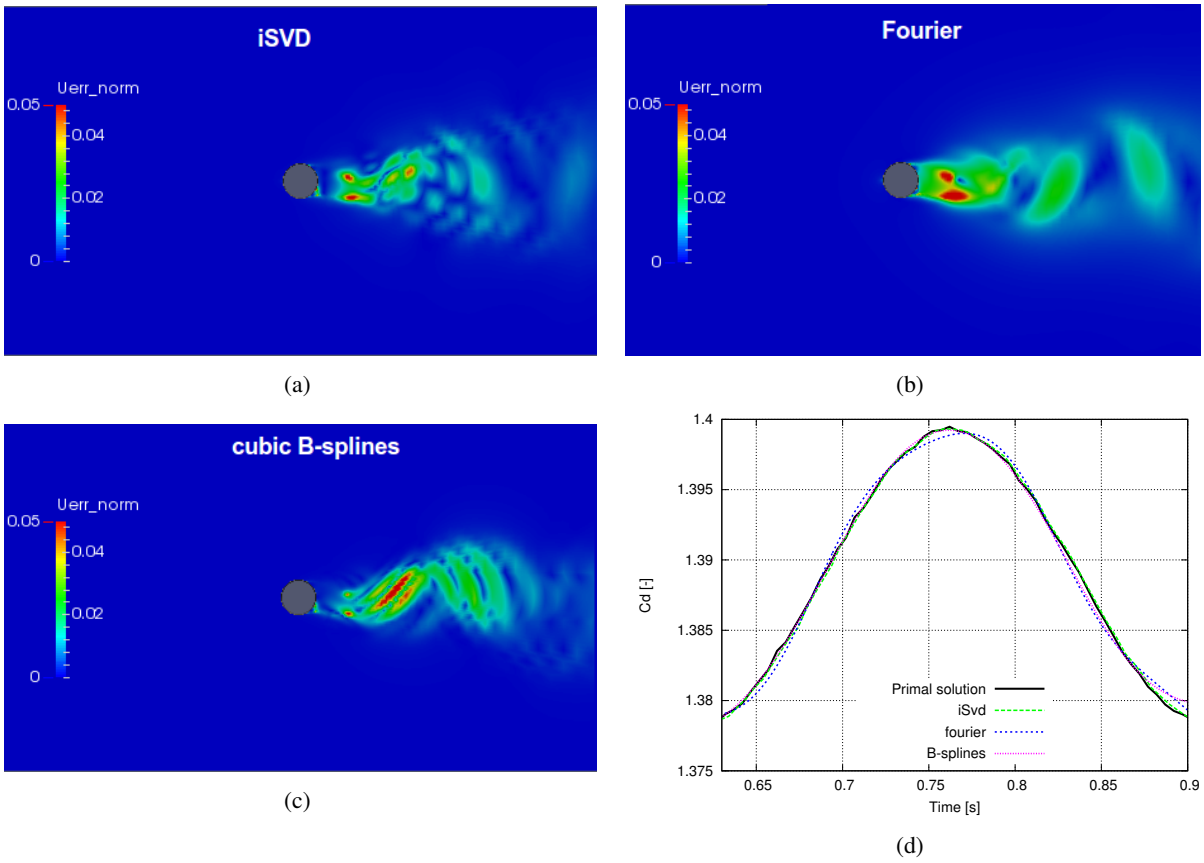


Figure 3: Flow around a cylinder. 15 snapshots are used to build all the approximation models. (a)-(c) The normalized error for the velocity field computed by each model for an arbitrary selected time step. (d) Comparison of the computed C_d time series by the three approximation models, after reconstructing the time-varying flow fields.

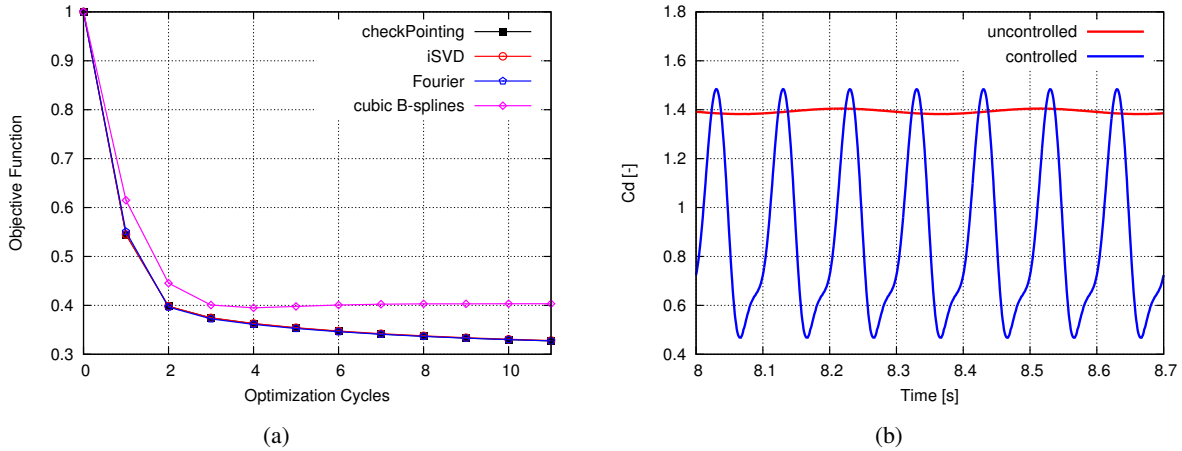
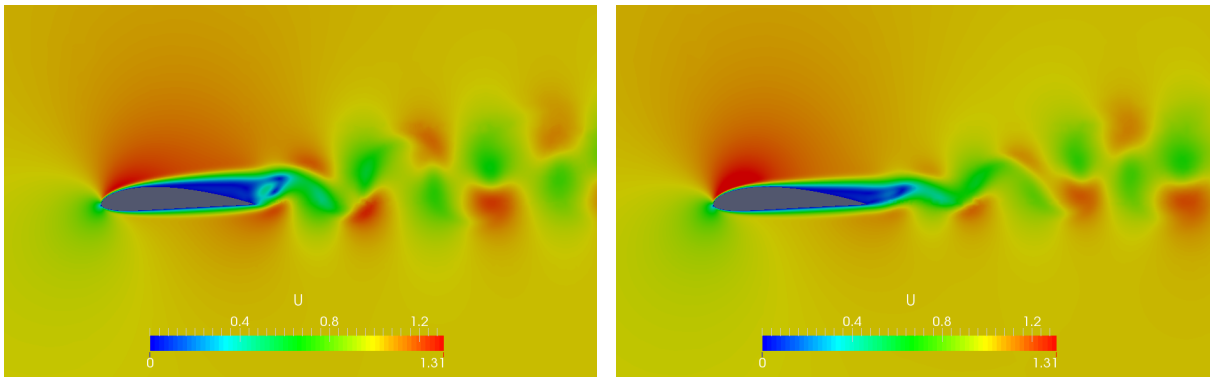


Figure 4: Flow around a cylinder. (a) Flow control optimization using approximations to the unsteady flow solution. The objective function (drag) is non-dimensionalized by its starting value. A steepest decent method is used in all cases. (b) Comparison for the computed drag coefficient, C_d, in the controlled and uncontrolled case.

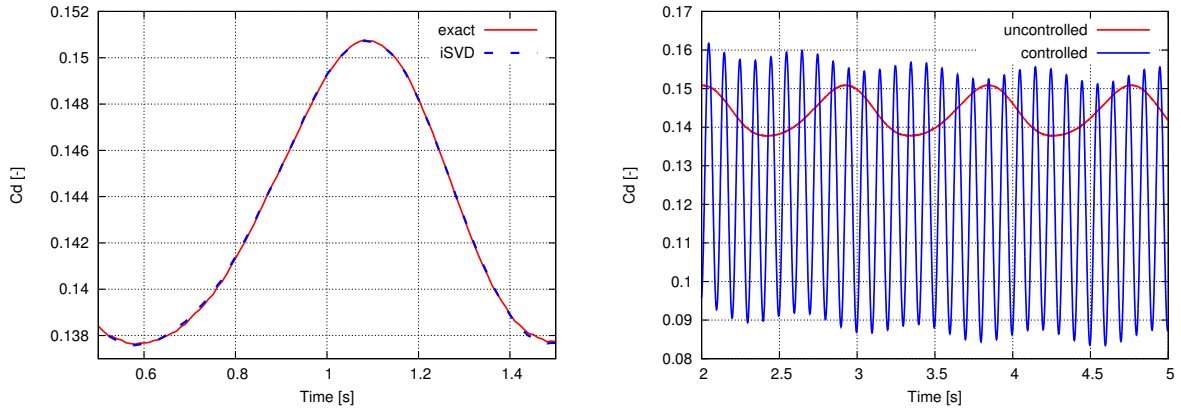
cells. We aim at the optimal control of the flow developed around the cylinder, using pulsating jets to suppress the Karman vortices and the so-induced drag. The fixed jet positions can be seen in fig. 2a; their amplitudes are optimized according to the objective function of eq. (3).

The approximation models presented in the previous section are used to reconstruct the pressure and velocity fields during the numerical integration of the unsteady adjoint PDEs, backwards in time. Before focusing on the results



(a) (b)

Figure 5: Flow around NACA4415 airfoil. (a) The velocity field before the optimization. (b) The velocity field after the flow control optimization for the drag force minimization is applied.



(a) (b)

Figure 6: Flow around NACA4415 airfoil. (a) Comparison for the C_d as computed by the N-S solver and iSVD model when the truncation point is $k = 15$. (b) Comparison for the computed drag coefficient, C_d, in the uncontrolled and controlled case.

of the optimization, a comparison of the accuracy of the approximation models, on the uncontrolled case, based on a single period composed of 100 time-steps, is made. Fig. 3d shows the comparison of the computed drag coefficient C_d as this results by integrating the fields obtained from the numerical solution of the incompressible unsteady N-S equations and the interpolated/approximated ones. The same number of snapshots has been used for approximation models. Using the cubic B-splines and Fourier series models, some discrepancies can be observed. The iSVD-based ($k = 15$) model performs much better than the others. Regarding the iSVD model, this is due to the continuous readjustment of the orthogonal basis using the flow fields obtained at each time step. In fig. 3, a comparison of the normalized error for the velocity field, at the same instant, for all the approximation models is shown.

During the optimization process, ten periods of the flow phenomenon were simulated; the first and last four served to ensure that the transient primal and adjoint flows have been discarded and the other two, in the middle, are used for the computation of the objective function and the sensitivity derivatives. All the models used in this paper approximate the flow field evolution for six periods in total (those which the adjoint equations are solved for). An alternative approach would be to approximate a single period; however, it was decided not to do so, because the developed solver (based on the approximation models) is, among other, capable of handling optimization problems governed by transient/non-periodic flows. The initial jet amplitudes were set to zero.

The convergence of the optimization process based on approximation models for reconstructing the primal fields needed for the solution of the adjoint equations is shown in fig. 4a. It can be seen that the Fourier-based and iSVD models produce almost the same results with the exact model (binomial check-pointing). In the same figure, it is shown that the cubic B-splines model computes a slightly different minimum for the objective function. Fig. 4b

Table 1: Flow around a cylinder. Computed optimal jet amplitudes when the approximation models are used. Computed improvement (percentage) in the time-averaged drag. All three models are built on the basis of 15 snapshots.

Jet	Check-Pointing	Cubic B-splines	Fourier	iSVD
0	0.027738	0.031574	0.027824	0.027691
1	-0.039552	-0.020647	-0.040956	-0.038994
2	-0.126804	-0.076364	-0.126821	-0.126539
3	-0.085423	-0.064450	-0.082390	-0.086517
4	-0.017046	-0.038401	-0.015522	-0.018128
5	0.082419	0.061529	0.083470	0.081828
6	0.023486	-0.014347	0.020932	0.023401
Objective function change (percentage)				
	–67.25%	–59.67%	–67.26%	–67.19%

shows the C_d time-series computed after the optimization process using the iSVD model; the Fourier series model could be used instead since it produces results which are almost identical to the ones of the iSVD. In the controlled case, the C_d time-series, even though larger in amplitude, gives a smaller mean C_d of ~ 1 in contrast to ~ 1.39 of the uncontrolled case. Table 1 lists the optimal amplitude for each (pair of) jet(s). From the tabulated results, it is seen that the cubic B-splines model computed a different sign for the seventh jet, while non-negligible differences for all jet amplitudes are observed.

The second case is concerned with the control of the flow around the NACA4415 isolated airfoil using four pulsating jets. The hybrid computational mesh has $\sim 144 \cdot 10^3$ cells, composed of quadrilateral elements close to the wall and triangular ones elsewhere. The angle of attack has been set to 7.5^o and the flow is laminar with Re≈ 500. The same, as for the cylinder case, optimization is carried out (i.e. flow control by using pulsating jets), and the objective function is drag. The position of jets is seen in fig. 2b. The instantaneous velocity and pressure fields needed in the adjoint loop were provided by the iSVD model with truncation at $k = 15$. As shown in fig. 6a, the approximation error of this model is very small. During the optimization process, the N-S solver run for 6 periods in order to ensure a periodic solution[1]. The time-step used was $\Delta t = 10^{-3} s$, which implies that 4000 snapshots were needed to compute the adjoint fields for four periods. In fig. 5a, the velocity field before the optimization is illustrated. The optimization process resulted in the velocity field shown in fig. 5b. It can be seen that the separation over the airfoil suction side is smaller when the optimal flow control is activated, compared to the one seen for the uncontrolled airfoil. However, it should be noted that the Karman vortex street is not completely eliminated. Fig. 6b presents a comparison for the C_d before and after the optimization. For the controlled case, the time series for the computed C_d exhibits a dominant frequency which is the one jets operate with (i.e. $f^n = 10 Hz$); however, other frequencies exist too. This is an indication for the persistent, though reduced, Karman vortex street shown in fig. 5b. The mean value of the C_d reduced to ~ 0.12 due to the application of flow control, while in the uncontrolled case the mean value was ~ 0.145. The optimal jet amplitudes can be found in table 2.

Table 2: Flow around the NACA4415 airfoil. Optimal jet amplitudes computed by the optimization process.

Jet	Amplitude
0	-0.064655
1	-0.085017
2	-0.101644
3	-0.115432

6 DISCUSSION-CONCLUSIONS

When the optimization problem involves unsteady flow phenomena, the apparent choice is to use the unsteady adjoint method (in its continuous or discrete variant) to compute the gradient of the objective function and support a gradient-based method. However, this requires the instantaneous flow fields (e.g. pressure, velocities) to be avail-

[1] No transient flow residue in the periods used for the computations of the objective function and the sensitivity derivatives.

able at each time instant. This may lead to non-affordable storage, exceeding even the computational resources. Approximation models which are used to reconstruct the instantaneous flow fields while marching backwards in time can be used instead. Though less accurate, these models have less memory requirements. In this paper, for approximation/interpolation purposes, cubic B-splines, Fourier series and incremental SVD models have been tested.

The outcome of the optimization process using approximation models presents negligible differences, for most of the models, compared to the exact solution. In terms of computational cost[2], in both cases, the results obtained showed that all the optimization processes relying upon approximation models run faster compared to binomial check-pointing. For instance, for the cylinder case[3], the cubic B-splines model needed $\sim 42\%$ of the computational cost of the check-pointing technique, Fourier series model needed $\sim 47\%$, whereas the iSVD model needed $\sim 56\%$.

In conclusion, approximation models, such as Fourier series and iSVD, are viable techniques to be used in the framework of unsteady continuous adjoint based optimization. Being properly tuned (e.g. selecting the right truncation point in iSVD model) could reduce significantly the memory cost (i.e. to $\sim 2.5\%$ for the cylinder and $\sim 3.75\%$ for the NACA4415 cases) compared to full storage.

ACKNOWLEDGMENT

This research was funded from the People Programme (ITN Marie Curie Actions) of the European Union's 7^{th} Framework Programme (FP7/2007-2013) under REA Grant Agreement n^o 317006 (AboutFLOW project). The second author is an AboutFLOW Early Stage Researcher.

References

[1] M. Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra and its Applications*, 415(1):20 – 30, 2006.

[2] A. Carnarius, F. Thiele, Özkaya E., A. Nemili, and N. Gauger. Optimal control of unsteady flows using a discrete and a continuous adjoint approach. *System Modeling and Optimization, Advances in Information and Communication Technology*, 391:318–327, 2011.

[3] A. Griewank and A. Walther. Algorithm 799: Revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Trans. on Math. Software (TOMS)*, 26(1):19–45, 2000.

[4] R. I. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, 62:40–65, 1986.

[5] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3:233–260, 1988.

[6] I. Kavvadias, G. Karpouzas, E. Papoutsis-Kiachagias, and K. Giannakoglou. Optimal flow control and topology optimization using the continuous adjoint method in unsteady flows. EUROGEN 2013, Gran Canaria, Spain, 2013.

[7] D. Papadimitriou and K. Giannakoglou. Aerodynamic shape optimization using first and second order adjoint and direct approaches. *Archives of Comp. Meth. in Eng.*, 15(4):447–488, 2008.

[8] E.M. Papoutsis-Kiachagias and K.C. Giannakoglou. Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications. *Arch. of Comput. Meth. in Eng. (ACME)*, pages 1–45, 2014.

[9] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64:97–110, 1974.

[10] C. Vezyris, I. Kavvadias, E. Papoutsis-Kiachagias, and K. Giannakoglou. Unsteady continuous adjoint method using pod for jet-based flow control. 11th World Congress on Computational Mechanics, ECCOMAS, Barcelona, Spain, 2014.

[11] Q. Wang, P. Moin, and G Iaccarino. Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation. *SIAM J. Sci. Comp.*, 31(4):25492567, 2008.

[2] Providing that the same memory is allocated and similar results are computed by the optimization process

[3] Using 15 check-points, thus keeping the memory allocation constant for all the models

[12] A. Zymaris, D. Papadimitriou, E. Papoutsis-Kiachagias, K. Giannakoglou, and C. Othmer. The continuous adjoint method as a guide for the design of flow control systems based on jets. *Eng. Comp.*, 30(4), 2013.