

# Using coupled POD reduced models in a Trust-Region method for flow control

*M. Samuelides<sup>†\*</sup> and E. Juliani<sup>†\*</sup>*

*\*ONERA-Toulouse Research Center*

*2 Avenue Edouard Belin, BP 4025, 31055 Toulouse Cedex 4, France*

*†SUPAERO*

*10 Avenue Edouard Belin, BP 4032, 31055 Toulouse Cedex, France*

## Abstract

Optimal flow control problems are often hard to solve directly. One has to face high dimensionality when numerically simulating the solution in an optimization loop. A possible way to overcome computational complexity consists in projecting the state equation onto a relevant reduced basis which is obtained by extracting controlled state solution features. This methodology was introduced by Graham *et al.* [7]. It is called Proper Orthogonal Decomposition Reduced Order Modelling (POD-ROM) and leads to a simplified optimal control problem. The local accuracy of the POD-ROM around the generating control (as for quadratic approximations) leads us to design an iterative optimization algorithm with respect to the previous limitation. TRPOD algorithm was introduced by Arian *et al.* [1]. It is based upon a Trust-Region algorithm and successive Low-Order models, takes into account the locality property. However, a straight forward application of TRPOD is not sufficient to reach full convergence to an optimal solution.

We propose an improvement of the method which is based on the following principles :

- Corrections to the Low-Order model have to be made mainly in the decreasing direction of the original cost function.
- Former Low-Order models are interacting with the current one through a coupling energy instead of building a single joint model from all the past observations.

The resulting algorithm is then applied to the cylinder wake flow control problem in laminar regime. The flow is controlled by an harmonic angular velocity imposed to the cylinder's boundary. Numerical results (time computations and optimal control parameters) are compared to original TRPOD and BFGS approaches.

## 1 Notations

In this paper, we use a standardized notation, it means :

- $\mathbf{Z}$  is a vector of  $\mathbb{R}^2$  or vector value function.

- $Z$  is a scalar or scalar value function.
- $H^n(\Omega)$  is the set of vector value functions with components is in  $H^n(\Omega)$ .
- $(.,.)$  is inner scalar product of  $L^2$  or  $\mathbf{L}^2$ , with  $\|.\|$  as associated norm.
- $\|.\|_n$  is usual  $\mathbb{R}^n$  norm.

## 2 Flow Control Problem

### 2.1 Presentation and Governing equations

The physical problem that is introduced in this section deals with flow dynamics past a cylinder. Intensive investigations in an optimal control framework have been carried out for this configuration (see [12], [11], [17],[18]) so that explicit comparisons can be made throughout this article. Indeed, such a physical system happens to be relatively easy to be analysed or numerically simulated, so that a simple and portable code including both flow simulator and optimization algorithm can be built up.

The flow is supposed to be incompressible, with velocity denoted by  $\mathbf{Y}$ . The following parameters characterize the dynamics :

- The upstream flow velocity and pressure values  $(\mathbf{Y}_\infty, P_\infty)$ .
- Diameter  $D$  of the cylinder.
- Reynolds number  $Re$ , equal to 200 in this article.
- Frequency of vortex shedding  $f_s$ , governed by the Strouhal number of the vortex pattern

$$S_t = \frac{f_s D}{\|\mathbf{Y}_\infty\|_2}$$

- Aerodynamic coefficients

$$C_d \mathbf{e}_x + C_l \mathbf{e}_y = \int_{\Gamma_c} \left\{ \frac{P - P_\infty}{\frac{1}{2} \rho \|\mathbf{Y}_\infty\|_2^2} - \frac{2}{Re} \frac{\partial \mathbf{Y}}{\partial n} \right\} d\Gamma$$

where  $C_d$  and  $C_l$  are respectively the drag coefficient and the lift one,  $\Gamma_c$  is the cylinder boundary and  $\frac{\partial \mathbf{Y}}{\partial n}$  the normal partial derivative of  $\mathbf{Y}$  on the cylinder boundary.

For  $Re = 200$ , periodic fluctuations of the flow past the cylinder appear corresponding to the well known vortex shedding in laminar regime. The predominant frequency is equal to  $f_s$ . Flow oscillations lead to an increase of the drag coefficient  $C_d$ , compared to steady flow with the same Reynolds number. The resulting objective consists in reducing  $C_d$  by controlling the vortex shedding. A possible way is to build up an actuator enforced on the cylinder boundary. The control can be restricted to the following design :

$$\mathbf{U}(t, x) = \gamma(t) \mathbf{t}(x)$$

where  $\mathbf{t}(x)$  is the vector tangent at  $x$  to the cylinder boundary.

Figure 1 shows how the control acts on the boundary.

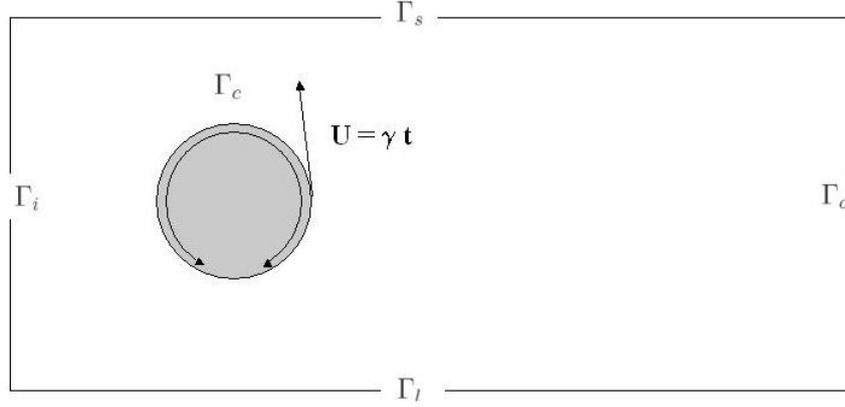


Figure 1: Configuration

The mathematical formulation of the previous problem is made through the introduction of 2-D unsteady Navier-Stokes equation:

$$\left\{ \begin{array}{l} \partial_t \mathbf{Y} - \frac{1}{Re} \Delta \mathbf{Y} + (\mathbf{Y} \cdot \nabla) \mathbf{Y} + \nabla P = 0 \quad \text{in } \Omega \times [0, T] \\ \nabla \cdot \mathbf{Y} = 0 \quad \text{in } \Omega \times [0, T] \\ \mathbf{Y} = \mathbf{U} \quad \text{on } \Gamma_c \times [0, T] \\ \mathbf{Y}(t=0) = \mathbf{Y}_0 \\ (\mathbf{Y}(t, |x| \rightarrow +\infty), P(t, |x| \rightarrow +\infty)) = (\mathbf{Y}_\infty, P_\infty) \end{array} \right.$$

where  $\mathbf{Y}_0$  is a snapshot of the control-free flow.

Since  $\nabla \cdot \mathbf{Y} = 0$ , one has the following equality

$$(\mathbf{Y} \cdot \nabla) \mathbf{Y} = (\mathbf{Y} \cdot \nabla) \mathbf{Y} + (\nabla \cdot \mathbf{Y}) \mathbf{Y} \equiv \nabla \cdot (\mathbf{Y} \otimes \mathbf{Y})$$

which is used in the next sections to implement the solver. In the sequel  $(\mathbf{Y} \cdot \nabla) \mathbf{Y}$  is replaced by the right hand part

$$\nabla \cdot (\mathbf{Y} \otimes \mathbf{Y})$$

For numerical purposes, a bounded domain  $\Omega$  including the cylinder is defined as shown in Figure 1. Nevertheless flow behaviour in  $\Omega$  must remain as close as possible to the unbounded case. Therefore an inflow boundary condition of is introduced at  $\Gamma_i$ ,  $\Gamma_s$ ,  $\Gamma_l$ :

$$\mathbf{Y} = \mathbf{Y}_\infty$$

An outflow boundary condition of the form

$$\partial_n \mathbf{Y} - P \mathbf{n} = 0$$

is applied to the right boundary  $\Gamma_o$ .

The resulting Navier-Stokes equation used for numerical simulation is:

$$\begin{aligned}
\partial_t \mathbf{Y} - \frac{1}{Re} \Delta \mathbf{Y} + \nabla \cdot (\mathbf{Y} \otimes \mathbf{Y}) + \nabla P &= 0 \quad \text{in } \Omega \times [0, T] \\
\nabla \cdot \mathbf{Y} &= 0 \quad \text{in } \Omega \times [0, T] \\
\mathbf{Y} &= \mathbf{Y}_\infty \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \times [0, T] \\
\partial_n \mathbf{Y} - P \mathbf{n} &= 0 \quad \text{on } \Gamma_o \times [0, T] \\
\mathbf{Y} &= \gamma(t) \mathbf{t}(\mathbf{x}) \quad \text{on } \partial \Gamma_c \times [0, T] \\
\mathbf{Y}(t=0) &= \mathbf{Y}_0 \quad \text{in } \Omega
\end{aligned} \tag{1}$$

## 2.2 Optimal Control Problem formulation

Recall that we are interested in reducing the drag coefficient when facing a nonstationary wake flow behind the cylinder with  $Re = 200$ . This can be formulated in terms of Optimal Control Problem, where the desired angular velocity  $\bar{\gamma}$  ( $\mathbf{U} = \bar{\gamma} \mathbf{t}$ ) is a solution of the optimization problem:

$$\bar{\gamma} = \operatorname{argmin}_\gamma J(\gamma, \mathbf{Y})$$

where  $J$  is a cost function and  $(\mathbf{Y}, P)$  satisfies eq. (1).

Several formulations of  $J$  are suggested in the litterature and briefly described by [12].

### Drag coefficient minimization.

According to [11],[10], the cost function formulation introduces the drag work through the viscous dissipation energy and is defined by:

$$J(\gamma, \mathbf{Y}) = \frac{2}{Re} \int_0^T \int_\Omega \left( \frac{1}{2} |\nabla \mathbf{Y} + \nabla \mathbf{Y}^T| \right)^2 d\Omega dt$$

where  $\frac{1}{2}(\nabla \mathbf{Y} + \nabla \mathbf{Y}^T)$  is the rate of deformation tensor. Minimizing  $J$  consists in finding a rotation  $\gamma$  that would reduce the viscous dissipation energy and therefore the drag coefficient.

### Enstrophy minimization.

Since  $\omega(\mathbf{Y}, t) = \partial_y Y_1 - \partial_x Y_2$  is the vorticity quantity (where  $\mathbf{Y} = (Y_1, Y_2)$ ), we define  $J$  as:

$$J(\gamma, \mathbf{Y}) = \frac{1}{Re} \int_0^T \|\omega(\mathbf{Y}, t)\|^2 dt$$

### Flow tracking

A desired flow profile without vortices is defined with velocity and pressure fields  $(\mathbf{Y}_d, P_d)$ . The cost function is defined by:

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \int_0^T \|\mathbf{Y} - \mathbf{Y}_d\|^2 dt + \frac{\beta}{2} \|\mathbf{Y}(T) - \mathbf{Y}_d(T)\|^2 \tag{2}$$

Minimizing  $J$  consists in finding  $\bar{\gamma}$  so that the controlled flow  $\bar{\mathbf{Y}}$  is as close as possible to  $\mathbf{Y}_d$ . The resulting drag coefficient is expected to be reduced relatively to the uncontrolled one. Choosing the desired flow depends on authors:

- [17] suggest to use the corresponding unstable stationary flow for the same Reynolds number ( $Re_d = 200$ ).
- [12] use a flow profile obtained with a low Reynolds number ( $Re_d = 2$ ).

In the sequel, flow tracking type cost function is only considered. According to [12] the best results were reached for such a formulation of  $J$ . A discussion about what flow profile  $Y_d$  to choose is made in the result section. For  $Re_d = 30$  (close to the limit Reynolds number for which instabilities appear) and  $\beta = 0$  our numerical results are similar to the ones presented in [11].

### Cost function Regularization

As mentioned in [12], dealing with (2) doesn't ensure the well-posedness of the optimal control system. Numerically, we observe that a small decrease of the cost function is associated with a large variation of the control parameter. Therefore the sub-optimal control obtained numerically is not expected to be a good approximation of the optimal control.

In order to avoid such a phenomena, a Tikhonov-Arsenin penalization involving the control parameter is added to the cost function:

$$\tilde{J}(\gamma, \mathbf{Y}) = J(\gamma, \mathbf{Y}) + \delta \Pi$$

Where  $\Pi$  is has the following form:

$$\Pi(\gamma) = \frac{1}{2} \int_0^T \gamma^2(t) dt$$

The final cost function to be minimized in the sequel is:

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \int_0^T \|\mathbf{Y} - \mathbf{Y}_d\|^2 dt + \frac{\delta}{2} \int_0^T \gamma^2(t) dt \quad (3)$$

## 2.3 Cost function Gradient

Recall that we want to minimize (3) subject to Navier-Stokes equation (1) as state constraint.

Basic algorithms for minimizing the functional  $F(\gamma) = J(\gamma, \mathbf{Y}_\gamma)$  (where  $\mathbf{Y}_\gamma$  is solution of the previous state equation (1)) are based upon gradient evaluation against direction  $\gamma_2$ :

$$\langle \nabla F(\gamma), \gamma_2 \rangle \equiv \int_0^T \nabla F(\gamma) \gamma_2 dt$$

We have the following expression:

$$\langle \nabla F(\gamma), \gamma_2 \rangle = \int_0^T (\mathbf{Y} - \mathbf{Y}_d, \mathbf{Z}) dt + \delta \int_0^T \gamma \gamma_2 dt$$

where  $\mathbf{Z}$  satisfies

$$\begin{aligned}
 \partial_t \mathbf{Z} - \frac{1}{Re} \Delta \mathbf{Z} + \nabla \cdot (\mathbf{Y} \otimes \mathbf{Z}) + \nabla \cdot (\mathbf{Z} \otimes \mathbf{Y}) + \nabla Q &= 0 & \text{in } \Omega \times [0, T] \\
 \nabla \cdot \mathbf{Z} &= 0 & \text{in } \Omega \times [0, T] \\
 \mathbf{Z} &= 0 & \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \times [0, T] \\
 \partial_n \mathbf{Z} - Q \mathbf{n} &= 0 & \text{on } \Gamma_o \times [0, T] \\
 \mathbf{Z} &= \gamma_2 \mathbf{t} & \text{on } \Gamma_c \times [0, T] \\
 \mathbf{Z}(t=0) &= 0 & \text{in } \Omega
 \end{aligned} \tag{4}$$

Equation (4) is linear and forward in time. Moreover the computation of  $Z$  at time  $t$  only depends on the previous time values of  $\mathbf{Y}$ . However, identifying  $\nabla F(\gamma)$  would lead to implement  $Z$  in every admissible direction  $\gamma_2$ , which could be unfeasible. In order to have an intrinsic formulation of the gradient, the adjoint formulation is used instead:

$$\nabla F(\gamma) = -(\partial_n \mathbf{Z} - Q \mathbf{n})_{|\Gamma_c}$$

where  $(\mathbf{Z}, Q)$  satisfies

$$\begin{aligned}
 -\partial_t \mathbf{Z} - \frac{1}{Re} \Delta \mathbf{Z} + \nabla \cdot (\mathbf{Y} \otimes \mathbf{Z}) + \nabla \cdot (\mathbf{Z} \otimes \mathbf{Y}) + \nabla Q &= \mathbf{Y} - \mathbf{Y}_d & \text{in } \Omega \times [0, T] \\
 \nabla \cdot \mathbf{Z} &= 0 & \text{in } \Omega \times [0, T] \\
 \mathbf{Z} &= 0 & \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \times [0, T] \\
 \partial_n \mathbf{Z} - Q \mathbf{n} &= 0 & \text{on } \Gamma_o \times [0, T] \\
 \mathbf{Z} &= 0 & \text{on } \Gamma_c \times [0, T] \\
 \mathbf{Z}(t=T) &= 0 & \text{in } \Omega
 \end{aligned} \tag{5}$$

The adjoint equation is linear and backward in time. Therefore  $\mathbf{Y}$  has to be first solved before computing  $\mathbf{Z}$ .

## 3 Discretization

### 3.1 Discretization of the State Equation

#### 3.1.1 Time discretization

Simulating the flow behaviour past a cylinder is relatively simple. Classical mathematical tools are used for numerically solving the unstationary Navier-Stokes equation including a control parameter.

We define  $T$  as the length of the time window,  $\Delta t$  the time step and  $n_t = \frac{T}{\Delta t}$ .

A semi-implicit time discretization is considered, linearizing the convection part of the state equation. For  $1 \leq i \leq n_t - 1$ , without divergence-free constraint, velocity and pressure fields are updated according to the following equation:

$$\begin{aligned}
 \frac{1}{\Delta t} (\mathbf{Y}^{i+1} - \mathbf{Y}^i) - \frac{1}{Re} \Delta \mathbf{Y}^{i+1} + \nabla \cdot (\mathbf{Y}^{i+1} \otimes \mathbf{Y}^i) + \nabla P^{i+1} &= 0 & \text{in } \Omega \\
 \nabla \cdot \mathbf{Y}^{i+1} &= 0 & \text{in } \Omega \\
 \mathbf{Y}^{i+1} &= \mathbf{Y}_\infty & \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \\
 \partial_n \mathbf{Y}^{i+1} - P^{i+1} \mathbf{n} &= 0 & \text{on } \Gamma_o \\
 \mathbf{Y}^{i+1} &= \gamma((i+1)\Delta t) \mathbf{t} & \text{on } \Gamma_c
 \end{aligned} \tag{6}$$

### 3.1.2 Prediction-Correction method : Presentation

The main issue consists in solving the state equation with respect to the divergence-free constraint imposed to the state solution. A possible way to overcome this difficulty is based upon the Hodge decomposition theorem, assessing that every function  $\mathbf{Y}^*$  can be split according to the following equality :

$$\mathbf{Y}^* = \mathbf{Y} + \nabla\theta$$

where  $\mathbf{Y}$  is divergence-free and  $\theta$  a potential function. This leads to

$$\mathbf{Y} = \mathbf{Y}^* - \nabla\theta \quad (7)$$

Projection method introduced by Chorin [4] is based upon the previous property and a prediction-correction strategy. At each timestep,  $(\mathbf{Y}^i, P^i)$  is predicted by  $(\mathbf{Y}^{*i}, P^{*i})$  where the latter doesn't respect divergence-free constraint. A correction is then applied to  $(\mathbf{Y}^{*i}, P^{*i})$  by adding the  $\nabla\theta$  part, where  $\theta$  satisfies the Heat Equation. Since, this strategy has been developed and improved (e.g [8, 19, 15, 14]). The solver we implemented was introduced by Bergmann [2] and Braza [13], using a pressure-correction principle.

#### Prediction.

For  $1 \leq i \leq N$ , a prediction  $\mathbf{Y}^{*i+1}$  of velocity field  $\mathbf{Y}^{i+1}$  is computed by solving the following equation:

$$\begin{aligned} \frac{1}{\Delta t} (\mathbf{Y}^{*i+1} - \mathbf{Y}^i) - \frac{1}{Re} \Delta \mathbf{Y}^{*i+1} + \nabla \cdot (\mathbf{Y}^{*i+1} \otimes \mathbf{Y}^i) + \nabla P^{*i+1} &= 0 \quad \text{in } \Omega \\ \mathbf{Y}^{*i+1} &= \mathbf{Y}_\infty \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \\ \partial_n \mathbf{Y}^{*i+1} - P^{*i+1} \mathbf{n} &= 0 \quad \text{on } \Gamma_o \\ \mathbf{Y}^{*i+1} &= \gamma^i \mathbf{t} \quad \text{on } \Gamma_c \end{aligned} \quad (8)$$

where  $P^{*i+1}$  is pressure field prediction,  $(\gamma^i)_i$  is a discret approximation of the rotation values. As pressure plays the role of a Lagrangian parameter and cannot be explicitly computed, we simply choose  $P^{*i+1} = P^i$ .

#### Heat equation.

In order to update velocity field according to equality (7), we have to find the equation satisfied by  $\theta$ . By taking the divergence of (7), one has

$$\Delta\theta = \nabla \cdot \mathbf{Y}^{*i+1}$$

By noticing that  $\mathbf{Y}^{*i+1} = \mathbf{Y}^{i+1}$  on  $\Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c$ , we obtain

$$\frac{\partial\theta}{\partial\mathbf{n}} = 0 \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c$$

For stability reasons an homogeneous dirichlet boundary condition is enforced on the right boundary  $\Gamma_o$  :

$$\theta = 0$$

Finally,  $\theta$  is solution to equation:

$$\begin{aligned} \Delta\theta &= \nabla \cdot \mathbf{Y}^{*i+1} \\ \partial_n \theta &= 0 \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c \\ \theta &= 0 \quad \text{on } \Gamma_o \end{aligned} \quad (9)$$

**Correction.**

Velocity field is updated using equality (7):

$$\mathbf{Y}^{i+1} = \mathbf{Y}^{*i+1} - \nabla\theta \quad (10)$$

If  $\mathbf{Y}^{i+1}$  is replaced by  $\mathbf{Y}^{*i+1} - \nabla\theta$  in (6), and if (8) is subtracted to the result, we obtain

$$\nabla P^{i+1} = \nabla P^i + \frac{1}{\Delta t} \nabla\theta + \nabla \cdot (\nabla\theta \otimes \mathbf{Y}^i) - \frac{1}{Re} \Delta \nabla\theta$$

By neglecting the two last terms, we obtain

$$P^{i+1} = P^i + \frac{1}{\Delta t} \theta \quad (11)$$

**3.1.3 Spatial discretization**

Finite Element Method is used for spatial discretization. Variational formulation is obtained for each of the three previous steps of the solver (Prediction, Heat Equation, Correction). We follow notations and results from Guermond [9] in the sequel.

According to the boundary conditions of equations (8) and (9), we define

$$X_0 = \{ \mathbf{Y} \in \mathbf{H}^1(\Omega) \mid \mathbf{Y}|_{\Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c} = 0 \}$$

and

$$N_0 = \{ P \in H^1(\Omega) \mid P|_{\Gamma_o} = 0 \}$$

We assume that

$$\begin{aligned} \exists \hat{\mathbf{t}} \in \mathbf{H}^1(\Omega) \quad & \hat{\mathbf{t}}|_{\Gamma_c} = \mathbf{t}(x), \hat{\mathbf{t}}|_{\Gamma_i \cup \Gamma_s \cup \Gamma_l} = 0, \partial_n \hat{\mathbf{t}}|_{\Gamma_o} = 0 \\ \exists \hat{\mathbf{Y}}_\infty \in \mathbf{H}^1(\Omega) \quad & \hat{\mathbf{Y}}_\infty|_{\Gamma_i \cup \Gamma_s \cup \Gamma_l} = \mathbf{Y}_\infty, \hat{\mathbf{Y}}_\infty|_{\Gamma_c} = 0, \partial_n \hat{\mathbf{Y}}_\infty|_{\Gamma_o} = 0 \end{aligned} \quad (12)$$

where we recall that  $\mathbf{t}(x)$  is the vector tangent at  $x$  to the cylinder boundary.

Consider

$$\hat{\mathbf{Y}}^{*i+1} = \mathbf{Y}^{*i+1} - \hat{\mathbf{Y}}_\infty - \gamma^i \hat{\mathbf{t}}$$

that satisfies

$$\begin{aligned} \frac{1}{\Delta t} \left( \hat{\mathbf{Y}}^{*i+1} - \mathbf{Y}^i \right) - \frac{1}{Re} \Delta \hat{\mathbf{Y}}^{*i+1} + \nabla \cdot (\hat{\mathbf{Y}}^{*i+1} \otimes \mathbf{Y}^i) + \nabla P^{*i+1} &= f \quad \text{in } \Omega \\ \hat{\mathbf{Y}}^{*i+1} &= 0 \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c \\ \partial_n \hat{\mathbf{Y}}^{*i+1} - P^{*i+1} \mathbf{n} &= 0 \quad \text{on } \Gamma_o \end{aligned} \quad (13)$$

where

$$f = -\frac{1}{\Delta t} \left( \hat{\mathbf{Y}}_\infty + \gamma^i \hat{\mathbf{t}} \right) + \frac{1}{Re} \Delta (\gamma^i \hat{\mathbf{t}}) - \nabla \cdot ((\hat{\mathbf{Y}}_\infty + \gamma^i \hat{\mathbf{t}}) \otimes \mathbf{Y}^i) \quad \text{in } \Omega$$

**Variational formulation of (13).**

$\forall \mathbf{Z} \in X_0$ , we have

$$\left( \frac{\hat{\mathbf{Y}}^{*i+1} - \mathbf{Y}^i}{\Delta t}, \mathbf{Z} \right) + a(\hat{\mathbf{Y}}^{*i+1}, \mathbf{Z}) + b(\mathbf{Y}^i, \hat{\mathbf{Y}}^{*i+1}, \mathbf{Z}) + (\nabla P^{*i+1}, \mathbf{Z}) = l(\mathbf{Z}) \quad (14)$$

where

$$\begin{aligned} l(\mathbf{Z}) &= -\frac{1}{\Delta t} (\mathbf{Y}_\infty + \gamma^i \hat{\mathbf{t}}, \mathbf{Z}) - \gamma^i a(\hat{\mathbf{t}}, \mathbf{Z}) - b(\mathbf{Y}^i, \mathbf{Y}_\infty, \mathbf{Z}) - \gamma^i b(\mathbf{Y}^i, \hat{\mathbf{t}}, \mathbf{Z}) \\ a(\mathbf{Y}, \mathbf{Z}) &= \frac{1}{Re} (\nabla \mathbf{Y}, \nabla \mathbf{Z}) \\ b(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) &= ((\mathbf{X} \cdot \nabla) \mathbf{Y}, \mathbf{Z}) + (\nabla \cdot \mathbf{X}, \mathbf{Y} \cdot \mathbf{Z}) \end{aligned}$$

**Variational formulation of (9).**

$$\forall q \in N_0, (\nabla \theta, \nabla q) = - (\nabla \cdot \mathbf{Y}^{*i+1}, q) \quad (15)$$

**Variational formulation of (10)-(11).**

$$\begin{aligned} \forall \mathbf{Z} \in X_0, (\mathbf{Y}^{i+1}, \mathbf{Z}) &= (\mathbf{Y}^* + \nabla \theta, \mathbf{Z}) \\ \forall q \in N_0, (\nabla \theta, \nabla q) &= - (\nabla \cdot \mathbf{Y}^{*i+1}, q) \end{aligned} \quad (16)$$

**Finite element approximation.**

In order to be numerically feasible,  $X_0$  and  $N_0$  have to be replaced by finite dimensional versions, respectively denoted by  $X_{0h} \subset X_0$  and  $N_{0h} \subset N_0$ .

Several definitions of  $(X_{0h}, N_{0h})$  are available in the litterature:

- $(P_1, P_1)$  elements. Piecewise linear functions are used to generate both  $X_{0h}$  and  $N_{0h}$ . Unfortunately, this natural choice leads to pressure oscillations since Inf-Sup Stability Condition is not verified:

$$\exists \beta > 0, \inf_{Q_h \in N_{0,h}} \sup_{\mathbf{Y}_h \in X_{0,h}} (\nabla \cdot \mathbf{Y}_h, Q_h) \geq \beta \|\mathbf{Y}_h\|_{H^1} \|Q_h\|_{L^2}$$

- $(P_2, P_1)$  elements. Piecewise quadratic functions and piecewise linear ones are used to respectively define  $X_{0h}$  and  $N_{0h}$ . Inf-Sup Stability Condition is verified. However, this couple of elements is avoided because of heavy computing time cost.
- $(P_1/bubble, P_1)$  elements.  $N_{0h}$  is defined by  $P_1$  elements. In order to satisfy Inf-Sup condition, bubble functions are added to the initial  $P_1$ -elements basis in order to define  $X_{0h}$ . These elements are used to obtain spatial discretization.

## 3.2 Discretized Optimal Control Problem

The previous numerical scheme for solving state equation allows us to define a discretized version of cost function  $J$ :

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \sum_{i=0}^{n_t} \|\mathbf{Y}^i - \mathbf{Y}_d^i\|^2 \Delta t + \frac{\delta}{2} \sum_{i=0}^{n_t-1} (\gamma^i)^2 \Delta t \quad (17)$$

where  $\mathbf{Y} = (\mathbf{Y}^i)_{1 \leq i \leq n_t}$ .

### 3.3 Adjoint Equation

Two ways of computing the gradient of (17) are suggested in the litterature when it is based on a discret adjoint equation:

- Discretize both in time and space equation (5), with the same method used for Navier-Stokes equation.
- Compute the adjoint of the discret optimization system : minimize (17) subject to the discret equations (14)-(16) as state constraint.

The second choice is considered, as the computation of the discret adjoint equation is straightforward. Analysing equations (14)-(16), one can remark that formally

$$\begin{aligned} \mathbf{Y}^{i+1} &= A^i \mathbf{Y}^i + B^i P^i + C^i + \gamma^i D^i \\ P^{i+1} &= \tilde{A}^i \mathbf{Y}^i + \tilde{B}^i P^i + \tilde{C}^i + \gamma^i \tilde{D}^i \end{aligned} \quad (18)$$

The dependancy between  $(\mathbf{Y}^{i+1}, P^{i+1})$  and  $(\mathbf{Y}^i, P^i, \gamma^i)$  is affine.

We define the Lagrangian:

$$L((\gamma^i)_i, (\mathbf{Y}^i)_i, (P^i)_i, (\mathbf{Z}^i)_i, (Q^i)_i) = \begin{cases} \frac{1}{2} \sum_{i=0}^{n_t} \|\mathbf{Y}^i - \mathbf{Y}_d^i\|^2 \Delta t + \frac{\delta}{2} \sum_{i=0}^{n_t-1} (\gamma^i)^2 \Delta t \\ - \sum_{i=0}^{n_t-1} \left( \mathbf{Z}^{i+1}, \mathbf{Y}^{i+1} - A^i \mathbf{Y}^i - B^i P^i + C^i - \gamma^i D^i \right) \Delta t \\ - \sum_{i=0}^{n_t-1} \left( Q^{i+1}, P^{i+1} - \tilde{A}^i \mathbf{Y}^i - \tilde{B}^i P^i - \tilde{C}^i - \gamma^i \tilde{D}^i \right) \Delta t \end{cases} \quad (19)$$

Formally taking the derivatives according to  $\mathbf{Y}^i$ ,  $P^i$  and  $\gamma^i$ , this leads to the adjoint equations (20) and the cost function gradient (21)

$$\begin{aligned} \mathbf{Z}^i &= (A^i)^T \mathbf{Z}^{i+1} + (\tilde{A}^i)^T Q^{i+1} + (\mathbf{Y}^i - \mathbf{Y}_d^i) \\ \mathbf{Z}^{n_t} &= 0 \\ Q^i &= (B^i)^T \mathbf{Z}^{i+1} + (\tilde{B}^i)^T Q^{i+1} \\ Q^{n_t} &= 0 \end{aligned} \quad (20)$$

where  $(X)^T$  denotes the adjoint operator of  $X$ , and

$$(\nabla F(\gamma))_i = \delta \gamma^i + (D^i)^T \mathbf{Z}^{i+1} + (\tilde{D}^i)^T Q^{i+1} \quad (21)$$

## 4 Low Order Models

Optimization problems involving an equality constraint can be usually solved by two classes of algorithm, either deterministic algorithms, where at least a gradient information is needed in order to compute a descent direction, or stochastic algorithms such as genetic ones or simulated annealing for instance. In the case the cost function depends on a parameter that controls the behaviour of a PDE as equality constraint, the optimization problem discretization leads to the design of a large scale system. When an iterative optimization algorithm is defined, the PDE solution is repeatedly simulated. Therefore

time cost for solving the optimization problem becomes prohibitive, especially when real-time computation is required. A way to overcome the dimensional effect consists in setting up a Reduced Order Model of the controlled state equation, where the approximate state solution can be expressed in a low dimensional space. In the sequel, a reduced basis approach is considered and has to be found, on to which a Petrov-Galerkin Method would be made possible.

Several methods are available in the litterature for finding such a reduced basis, either by choosing a prior fixed one or by iteratively extracting it. In the first case, it can be choosen between Lagrange, Krylov, Hermite ones. In the latter case, the Proper Orthogonal Decomposition introduced by Lumley and used in an iterative sense by [1] is an example, and is considered in the next sections.

## 4.1 Proper Orthogonal Decomposition (POD)

### 4.1.1 Presentation

The POD approach was first developped by [3] in order to analyze unstationary turbulent flows, through extraction of key features. A set of vectors is obtained from flow snapshots, concentrating relevant information of the turbulent behaviour. In a more general framework, POD is also known as Principal Component Analysis or Karhunen Loeve Decomposition. The extraction is based on either the continuous in time flow velocity field or a set of discrete in time values of velocity field (called snapshots). The latter is considered for our POD decomposition.

Let's denote by  $\mathbf{Y}_m$  the mean of the velocity snapshots  $\{\mathbf{Y}^i, 1 \leq i \leq n_t\}$ :

$$\mathbf{Y}_m = \frac{1}{n_t} \sum_i \mathbf{Y}^i$$

Considering  $\hat{\mathbf{Y}}^i = \mathbf{Y}^i - \mathbf{Y}_m$  and  $r = \dim(\hat{\mathbf{Y}}^1, \dots, \hat{\mathbf{Y}}^{n_t})$ , the POD orthonormal basis  $(\boldsymbol{\psi}^i)_{1 \leq i \leq r}$  is solution to the following iterative optimization problem:

$$\forall i \in \{1 \dots r\}, \boldsymbol{\psi}^i = \underset{\|\boldsymbol{\psi}\|=1}{\operatorname{argmax}} \sum_{i=1}^{n_t} \left( \boldsymbol{\psi}, \hat{\mathbf{Y}}^i \right)^2, \boldsymbol{\psi} \perp \{\boldsymbol{\psi}^1 \dots \boldsymbol{\psi}^{i-1}\} \quad (22)$$

where we recall that  $(\cdot, \cdot)$  is the  $\mathbf{L}^2(\Omega)$  scalar product and  $\|\cdot\|$  the corresponding norm.

Considering

$$\begin{aligned} \Gamma : \mathbf{L}^2(\Omega) &\longrightarrow \mathbf{L}^2(\Omega) \\ \boldsymbol{\psi} &\longrightarrow \sum_{i=1}^{n_t} \left( \hat{\mathbf{Y}}^i, \boldsymbol{\psi} \right) \hat{\mathbf{Y}}^i \end{aligned}$$

Problem (22) is equivalent to:

$$\boldsymbol{\psi}^i = \underset{\|\boldsymbol{\psi}\|=1, \boldsymbol{\psi} \perp \{\boldsymbol{\psi}^1 \dots \boldsymbol{\psi}^{i-1}\}}{\operatorname{argmax}}_{\boldsymbol{\psi}} (\Gamma(\boldsymbol{\psi}), \boldsymbol{\psi})$$

Taking the Lagrangian

$$L(\boldsymbol{\psi}, l) = (\Gamma(\boldsymbol{\psi}), \boldsymbol{\psi}) + l(\|\boldsymbol{\psi}\|^2 - 1)$$

the first order optimality condition is

$$\forall \phi \perp \{\psi^1 \dots \psi^{i-1}\}, (\Gamma \psi^i, \phi) = l(\psi^i, \phi) \quad (23)$$

As  $\Gamma$  is a positive self adjoint Hilbert-Schmidt operator, it can be written in the form

$$\Gamma = \sum_{i=1}^r l_i (\mathbf{e}_i, \cdot) \mathbf{e}_i$$

where  $l_1 \geq l_2 \geq \dots \geq l_r > 0$  and  $(\mathbf{e}_i, \mathbf{e}_j) = \delta_{ij}$ .

Using (23) we can then iteratively prove that  $\psi^i \equiv \mathbf{e}_i$ , with  $(\Gamma(\psi^i), \psi^i) = l_i$ .

The relative information contained by the  $m$  first elements of the POD basis is defined by

$$I(m) = \frac{\sum_{i=1}^m l_i}{\sum_{i=1}^r l_i}$$

Finally, given a truncation parameter  $p$ , a reduced basis  $\{\psi^1, \dots, \psi^{n_p}\}$  is selected where  $n_p = \min\{j | I(j) \geq p\}$ .

#### 4.1.2 Implementation

Numerically, computing the eigenvectors of  $\Gamma$  would lead to the diagonalization of square matrix  $\sum_{i=1}^r \hat{\mathbf{Y}}^i (\hat{\mathbf{Y}}^i)^T M$ . This diagonalization becomes unfeasible when facing a large scale discrete system. In order to avoid this difficulty, the Singular Value Decomposition needs to be applied to a reasonably sized square matrix. Let's introduce

$$\begin{aligned} \tilde{\Gamma} : \mathbb{R}^{n_t} &\longrightarrow \mathbb{R}^{n_t} \\ (\alpha_i)_{1 \leq i \leq n_t} &\longrightarrow \left( \sum_{j=1}^{n_t} \alpha_j (\hat{\mathbf{Y}}^i, \hat{\mathbf{Y}}^j) \right)_{1 \leq i \leq n_t} \end{aligned}$$

where  $\tilde{\Gamma}$  is a positive self-adjoint Hilbert-Schmidt operator. If  $(\tilde{l}_i)_i$  is the decreasing sequence of  $\tilde{\Gamma}$  eigenvalues, we have

$$\forall i \in \{1 \dots r\}, l_i > 0 \Rightarrow \tilde{l}_i = l_i$$

Strictly positive eigenvectors of  $\Gamma$  can be retrieved thanks to the following equality:

$$\psi^i = \frac{1}{\sqrt{\tilde{l}_i}} \sum_{j=1}^{n_t} \tilde{\psi}^i(j) \hat{\mathbf{Y}}^j \quad (24)$$

where  $(\tilde{\psi}^i(j))_j$  is an eigenvector of  $\tilde{\Gamma}$  with eigenvalue  $\tilde{l}_i = l_i$ .

Usually  $n_t \ll n + m$  (recall that  $n, m$  are the number of P1 and bubble elements). Therefore, one would prefer to use  $\tilde{\Gamma}$  instead of  $\Gamma$  and would find the POD basis in three steps:

- Diagonalizing  $(\hat{\mathbf{Y}}^i, \hat{\mathbf{Y}}^j)_{1 \leq i, j \leq n_t}$ .

- Retrieving the POD basis thanks to formula (24).
- Selecting the first  $n_p$  POD elements according to the truncation criteria  $p$ .

Figure 2 illustrates the first POD elements obtained for control-free flow at  $Re = 200$ . We observe that the relevant features of vortex shedding past the cylinder is captured by the first POD elements.

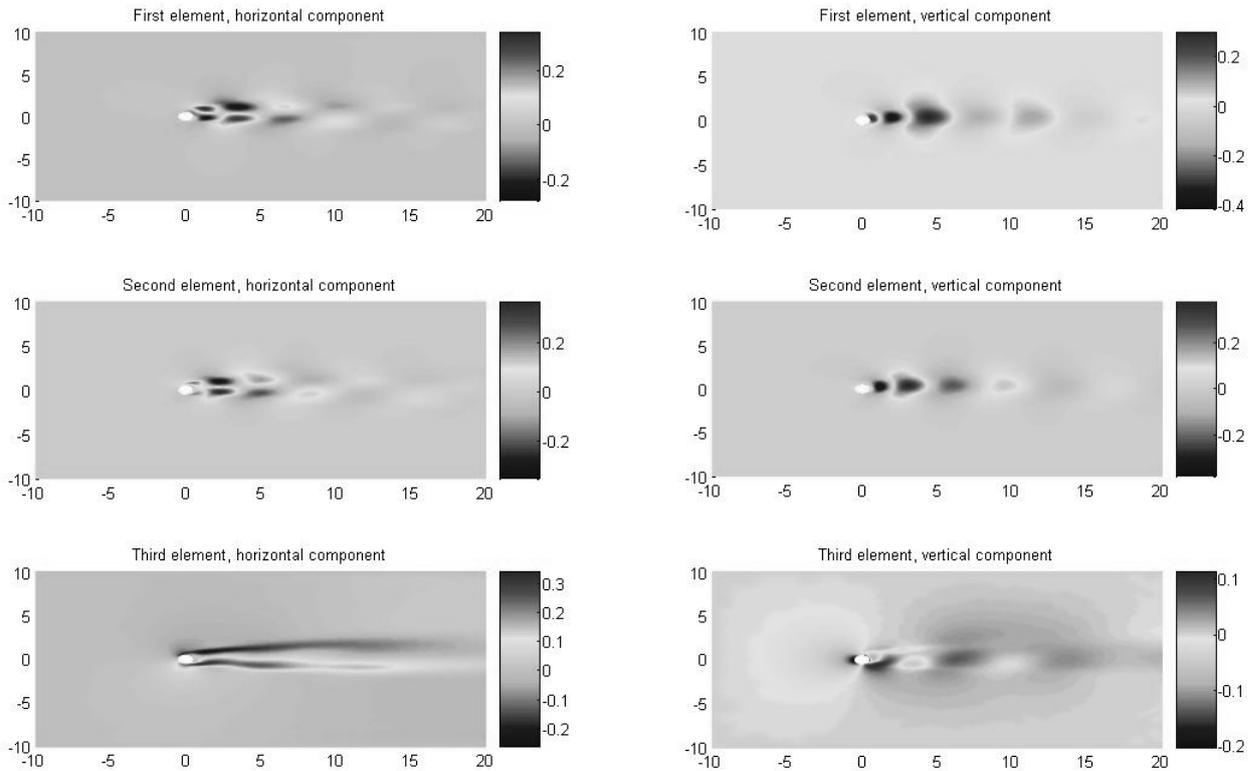


Figure 2: First POD functions

## 4.2 Reduced Order Model (POD-ROM)

The POD basis was first considered as a descriptive tool for turbulent flows. Since, it is used to generate a low order prediction model (also called POD-ROM) of the velocity field  $\mathbf{Y}$ . This prediction model is expected to fulfill both computational cost reduction and satisfactory approximation of  $\mathbf{Y}$  requirements. In the sequel, velocity prediction using POD-ROM is denoted by  $\mathbf{Y}^p$ .

### 4.2.1 Prediction in the control free case.

In control free case, prediction is obtained through a Galerkin projection of (1) onto the POD basis.

Recall that  $\boldsymbol{\psi}^i$  is expressed as a linear combination of the modified snapshots  $(\hat{\mathbf{Y}}^i)_{1 \leq i \leq nt}$ . Consequently,

$$\begin{aligned}\boldsymbol{\psi}_i &= 0 \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c \\ \nabla \cdot \boldsymbol{\psi}_i &= 0\end{aligned}$$

Define the velocity approximation by

$$\mathbf{Y}^p = \sum_{j=1}^{n_p} Y_j^p \boldsymbol{\psi}^j + \mathbf{Y}^m$$

As a prediction,  $\mathbf{Y}^p$  must satisfy (1).

Therefore, for every  $\boldsymbol{\psi}^i$ ,

$$\left( \boldsymbol{\psi}^i, \partial_t \mathbf{Y}^p - \frac{1}{Re} \Delta \mathbf{Y}^p + \nabla \cdot (\mathbf{Y}^p \otimes \mathbf{Y}^p) + \nabla P \right) = 0$$

Since

$$\begin{aligned}\frac{1}{Re} (\boldsymbol{\psi}^i, \Delta \mathbf{Y}^p) &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_{i,s,l,o,c}} (\boldsymbol{\psi}^i, \partial_n \mathbf{Y}^p)_2 d\Gamma \\ &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_o} (\boldsymbol{\psi}^i, \partial_n \mathbf{Y}^p)_2 d\Gamma \\ (\boldsymbol{\psi}^i, \nabla P) &= -(\nabla \cdot \boldsymbol{\psi}^i, P) + \int_{\Gamma_{i,s,l,o,c}} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma \\ &= \int_{\Gamma_o} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma \\ \nabla \cdot (\mathbf{Y}^p \otimes \mathbf{Y}^p) &= (\mathbf{Y}^p \cdot \nabla) \mathbf{Y}^p\end{aligned}$$

then

$$\begin{aligned}\frac{1}{Re} (\boldsymbol{\psi}^i, \Delta \mathbf{Y}^p) + (\boldsymbol{\psi}^i, \nabla P) &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_o} (\boldsymbol{\psi}^i, \partial_n \mathbf{Y}^p)_2 d\Gamma + \int_{\Gamma_o} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma \\ &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p)\end{aligned}$$

Finally,

$$(\boldsymbol{\psi}^i, \partial_t \mathbf{Y}^p + (\mathbf{Y}^p \cdot \nabla) \mathbf{Y}^p) + \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) = 0$$

In coefficient form:

$$\frac{dY_i^p}{dt} = a_i + \sum_{j=1}^{n_p} b_{ij} Y_j^p + \sum_{j=1}^{n_p} \sum_{k=1}^{n_p} c_{ijk} Y_j^p Y_k^p \quad (25)$$

where

$$\begin{aligned}a_i &= -(\boldsymbol{\psi}^i, (\mathbf{Y}^m \cdot \nabla) \mathbf{Y}^m) - \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^m) \\ b_{ij} &= -(\boldsymbol{\psi}^i, (\mathbf{Y}^m \cdot \nabla) \boldsymbol{\psi}^j) - (\boldsymbol{\psi}^i, (\boldsymbol{\psi}^j \cdot \nabla) \mathbf{Y}^m) - \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \boldsymbol{\psi}^j) \\ c_{ijk} &= -(\boldsymbol{\psi}^i, (\boldsymbol{\psi}^j \cdot \nabla) \boldsymbol{\psi}^k)\end{aligned}$$

### 4.2.2 Prediction in the controlled case.

Two predictions models were suggested by [7] in the controlled case:

- *Control Function Method.* The controlled part of the velocity field is predicted through a particular function denoted by  $\mathbf{Y}^c$ . It is computed before the optimization system. The velocity approximation is simply

$$\mathbf{Y}_\gamma^p = \sum_{j=1}^{n_p} Y_j^p \boldsymbol{\psi}_j + \mathbf{Y}^m + \gamma \mathbf{Y}^c$$

Snapshots are modified according to equality

$$\hat{\mathbf{Y}}^i = \mathbf{Y}^i - \mathbf{Y}^m - \gamma \mathbf{Y}^c$$

where  $\gamma$  is the cylinder rotation used to compute  $\{\mathbf{Y}^i, 1 \leq i \leq nt\}$  and  $\mathbf{Y}^m = \frac{1}{nt} \sum_i \mathbf{Y}^i$ . POD basis  $(\boldsymbol{\psi}^i)_{1 \leq i \leq r}$  obtained from the previous snapshots satisfies

$$\begin{aligned} \boldsymbol{\psi}^i &= 0 \quad \text{on} \quad \Gamma_i \cup \Gamma_s \cup \Gamma_l \cup \Gamma_c \\ \nabla \cdot \boldsymbol{\psi}^i &= 0 \end{aligned}$$

The Galerkin projection leads to a formulation similar to (25):

$$\frac{dY_i^p}{dt} = a_i + \sum_{j=1}^{n_p} b_{ij} Y_j^p + \sum_{j=1}^{n_p} \sum_{k=1}^{n_p} c_{ijk} Y_j^p Y_k^p + d_i \frac{d\gamma}{dt} + \left( e_i + \sum_{i=1}^{n_p} f_{ij} \right) \gamma + g_i \gamma^2 \quad (26)$$

where

$$\begin{aligned} d_i &= -(\boldsymbol{\psi}^i, \mathbf{Y}^c) \\ e_i &= -(\boldsymbol{\psi}^i, (\mathbf{Y}^m \cdot \nabla) \mathbf{Y}^c) - (\boldsymbol{\psi}^i, (\mathbf{Y}^c \cdot \nabla) \mathbf{Y}^m) - \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^c) \\ f_{ij} &= -(\boldsymbol{\psi}^i, (\boldsymbol{\psi}^j \cdot \nabla) \mathbf{Y}^c) - (\boldsymbol{\psi}^i, (\mathbf{Y}^c \cdot \nabla) \boldsymbol{\psi}^j) \\ g_i &= -(\boldsymbol{\psi}^i, (\mathbf{Y}^c \cdot \nabla) \mathbf{Y}^c) \end{aligned}$$

- *Penalty Method.* The original method for computing the POD basis is kept. The velocity approximation is

$$\mathbf{Y}^p = \sum_{j=1}^{n_p} Y_j^p \boldsymbol{\psi}^j + \mathbf{Y}^m$$

A penalized Robin boundary condition is enforced on the cylinder boundary in order to introduce the rotation  $\gamma$  in the variational formulation:

$$\begin{aligned} \partial_t \mathbf{Y}^p - \frac{1}{Re} \Delta \mathbf{Y}^p + \nabla \cdot (\mathbf{Y}^p \otimes \mathbf{Y}^p) + \nabla P &= 0 \quad \text{in} \quad \Omega \times [0, T] \\ \nabla \cdot \mathbf{Y}^p &= 0 \quad \text{in} \quad \Omega \times [0, T] \\ \mathbf{Y}^p &= \mathbf{Y}_\infty \quad \text{on} \quad \Gamma_i \cup \Gamma_s \cup \Gamma_l \times [0, T] \\ \partial_n \mathbf{Y}^p - P \mathbf{n} &= 0 \quad \text{on} \quad \Gamma_o \times [0, T] \\ \epsilon \partial_n \mathbf{Y}^p + \mathbf{Y}^p &= \gamma \mathbf{t} \quad \text{on} \quad \Gamma_c \times [0, T] \\ \mathbf{Y}^p(t=0) &= \mathbf{Y}_0 \quad \text{in} \quad \Omega \end{aligned} \quad (27)$$

For every  $\boldsymbol{\psi}^i$ ,

$$\left( \boldsymbol{\psi}^i, \partial_t \mathbf{Y}^p - \frac{1}{Re} \Delta \mathbf{Y}^p + \nabla \cdot (\mathbf{Y}^p \otimes \mathbf{Y}^p) + \nabla P \right) = 0$$

Since

$$\begin{aligned} \boldsymbol{\psi}^i &= 0 \quad \text{on } \Gamma_i \cup \Gamma_s \cup \Gamma_l \\ \nabla \cdot \boldsymbol{\psi}^i &= 0 \\ \boldsymbol{\psi}^i(t, x) &= g^i(t, x) \mathbf{t}(x) \quad \text{on } \Gamma_c \end{aligned}$$

we have

$$\begin{aligned} \frac{1}{Re} (\boldsymbol{\psi}^i, \Delta \mathbf{Y}^p) &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_{i,s,l,o,c}} (\boldsymbol{\psi}^i, \partial_n \mathbf{Y}^p)_2 d\Gamma \\ &= \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_o} (\boldsymbol{\psi}^i, \partial_n \mathbf{Y}^p)_2 d\Gamma - \frac{1}{Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \frac{\gamma \mathbf{t} - \mathbf{Y}^p}{\epsilon})_2 d\Gamma \\ (\boldsymbol{\psi}^i, \nabla P) &= - (\nabla \cdot \boldsymbol{\psi}^i, P) + \int_{\Gamma_{i,s,l,o,c}} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma \\ &= \int_{\Gamma_o} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma + \int_{\Gamma_c} (g^i \mathbf{t}, P \mathbf{n})_2 d\Gamma \\ &= \int_{\Gamma_o} (\boldsymbol{\psi}^i, P \mathbf{n})_2 d\Gamma \\ \nabla \cdot (\mathbf{Y}^p \otimes \mathbf{Y}^p) &= (\mathbf{Y}^p \cdot \nabla) \mathbf{Y}^p \end{aligned}$$

Therefore

$$\frac{1}{Re} (\boldsymbol{\psi}^i, \Delta \mathbf{Y}^p) + (\boldsymbol{\psi}^i, \nabla P) = \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \frac{\gamma \mathbf{t} - \mathbf{Y}^p}{\epsilon})_2 d\Gamma$$

Finally,

$$(\boldsymbol{\psi}^i, \partial_t \mathbf{Y}^p + (\mathbf{Y}^p \cdot \nabla) \mathbf{Y}^p) + \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^p) - \frac{1}{Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \frac{\gamma \mathbf{t} - \mathbf{Y}^p}{\epsilon})_2 d\Gamma = 0$$

In coefficient format:

$$\frac{dY_i^p}{dt} = a_i + \sum_{j=1}^{n_p} b_{ij} Y_j^p + \sum_{j=1}^{n_p} \sum_{k=1}^{n_p} c_{ijk} Y_j^p Y_k^p + d_i \gamma$$

where

$$\begin{aligned} a_i &= - (\boldsymbol{\psi}^i, (\mathbf{Y}^m \cdot \nabla) \mathbf{Y}^m) - \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \mathbf{Y}^m) - \frac{1}{\epsilon Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \mathbf{Y}^m)_2 d\Gamma \\ b_{ij} &= - (\boldsymbol{\psi}^i, (\mathbf{Y}^m \cdot \nabla) \boldsymbol{\psi}^j) - (\boldsymbol{\psi}^i, (\boldsymbol{\psi}^j \cdot \nabla) \mathbf{Y}^m) - \frac{1}{Re} (\nabla \boldsymbol{\psi}^i, \nabla \boldsymbol{\psi}^j) - \frac{1}{\epsilon Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \boldsymbol{\psi}^j)_2 d\Gamma \\ c_{ijk} &= - (\boldsymbol{\psi}^i, (\boldsymbol{\psi}^j \cdot \nabla) \boldsymbol{\psi}^k) \\ d_i &= \frac{1}{\epsilon Re} \int_{\Gamma_c} (\boldsymbol{\psi}^i, \mathbf{t})_2 d\Gamma \end{aligned}$$

In matrix format,

$$\frac{dY^p}{dt} = A + B Y^p + (C : \mathbf{Y}^p) Y^p + D \gamma \quad (28)$$

where

$$Y^p(t) = (Y_i^p(t))_{1 \leq i \leq n_p}$$

For adjoint equation computation simplicity, tensor  $C = (c_{ijk})_{ijk}$  is replaced by  $\tilde{C} = (\frac{1}{2}(c_{ijk} + c_{ikj}))_{ijk}$  in order to have the following symmetry:

$$\begin{aligned}\forall Y, Z, (\tilde{C} : Y)Z &= (\tilde{C} : Z)Y \\ (\tilde{C} : Y)Y &= (C : Y)Y\end{aligned}$$

For clarity reasons,  $\tilde{C}$  is simply denoted by  $C$ .

When admissible controls acting on the cylinder boundary are not restricted to rotations, the formulation of Control Function Method increases in complexity. We have to introduce a family of  $\{\mathbf{Y}^c\}$  in order to eliminate the control part of  $\mathbf{Y}$  and obtain an homogeneous Dirichlet boundary condition on  $\Gamma_c$ . Motivated by the generalization of our algorithm, the Penalty Method is only considered in the sequel.

### 4.3 Reduced Optimal Control Problem

#### 4.3.1 Formulation

In this section, a reduced model of the optimal control problem is suggested, using POD-ROM model. The matrix representation for the reduced POD-basis is  $\Gamma = (\boldsymbol{\psi}^1, \dots, \boldsymbol{\psi}^{n_p})$ , satisfying

$$\Gamma^T M \Gamma = I$$

where  $I$  is the identity matrix.

Recall that

$$\mathbf{Y}^p(t) = \sum_{j=1}^{n_p} Y_j^p(t) \boldsymbol{\psi}^j + \mathbf{Y}^m = \Gamma Y^p(t) + \mathbf{Y}^m$$

Assuming that  $\mathbf{Y}$  is replaced by his POD-ROM prediction, the Reduced Optimal Control problem takes the following form:

Minimize

$$\begin{aligned}J(\gamma, \mathbf{Y}^p) &= \frac{1}{2} \int_0^T \|\mathbf{Y}^p - \mathbf{Y}_d\|^2 dt + \frac{\delta}{2} \int_0^T (\gamma)^2 \\ &= \frac{1}{2} \int_0^T \|\Gamma Y^p + \mathbf{Y}^m - \mathbf{Y}_d\|^2 dt + \frac{\delta}{2} \int_0^T (\gamma)^2\end{aligned}$$

where  $Y^p$  is solution of (28).

#### 4.3.2 Discretization and Gradient computation

##### Discretization.

A semi-implicit scheme is used for time discretization of POD-ROM state equation. For  $1 \leq i \leq nt - 1$ ,

$$\frac{Y^{p,i+1} - Y^{p,i}}{\Delta t} = A + B Y^{p,i+1} + (C : Y^{p,i}) Y^{p,i+1} + D \gamma^i \quad (29)$$

The discretized version of the cost function is

$$J(\gamma, Y^p) = \frac{1}{2} \sum_0^T \|\Gamma Y^p + \mathbf{Y}^m - \mathbf{Y}_d\|^2 + \frac{\delta}{2} \sum_0^T \|\gamma\|^2$$

where

$$\begin{aligned}\sum_0^T \|\Gamma Y^p - \mathbf{Y}_d\| &= \sqrt{\sum_{i=0}^{nt-1} (\Gamma Y^{p,i} + \mathbf{Y}^m - \mathbf{Y}_d^i)^T M (\Gamma Y^{p,i} + \mathbf{Y}^m - \mathbf{Y}_d^i) \Delta t} \\ \sum_0^T \|\gamma\| &= \sqrt{\sum_{i=0}^{nt-1} (\gamma^i)^2 \Delta t}\end{aligned}$$

and  $Y^p$  is solution to (29).

### Gradient computation.

Adjoint equation of the discretized POD-ROM state equation is solved in order to compute the reduced cost function gradient. Let's introduce the Lagrangian

$$L(Y^p, \gamma, Z^p) = J(\gamma, Y^p) - \sum_{i=0}^{nt-1} (Z^{p,i+1})^T \left( \frac{Y^{p,i+1} - Y^{p,i}}{\Delta t} - A - B Y^{p,i+1} - (C : Y^{p,i}) Y^{p,i+1} - D \gamma^i \right)$$

First order optimality conditions for  $L$  leads to

$$\begin{aligned}\partial_Y L &= 0 \\ \partial_\gamma L &= 0 \\ \partial_Z L &= 0\end{aligned}$$

The adjoint equation is obtained from  $\partial_Y L = 0$ :

$$\begin{aligned}\frac{Z^{p,i} - Z^{p,i+1}}{\Delta t} &= B^T Z^{p,i} + (C : Y^{p,i-1})^T Z^{p,i} + (C : Y^{p,i+1})^T Z^{p,i+1} + Y^{p,i} + \Gamma^T M (\mathbf{Y}^m - \mathbf{Y}_d) \\ Z^{p,nt} &= 0\end{aligned} \quad (30)$$

As  $Y^p$  is controlled by  $\gamma$  (and written  $Y_\gamma^p$ ), the cost function depends only on the  $\gamma$  parameter. If  $F^p$  is introduced, where

$$F^p(\gamma) = J(\gamma, Y_\gamma^p)$$

we have

$$(\nabla_\gamma F^p)^i = \delta \gamma^i + D^T Z^{p,i+1} \quad (31)$$

by using optimality condition  $\partial_\gamma L = 0$ .

Computation of  $F^p$  value and gradient is based upon three steps:

- Compute  $Y^p$  thanks to (28).
- Compute  $Z^p$  thanks to (30).
- Compute  $\nabla F^p$  thanks to (31).

## 5 Trust Region Optimization

### 5.1 Trust Region-POD ([1])

Trust-Region methods are iterative optimization algorithms. The key idea consists in replacing the cost function by successive quadratic approximations and in optimizing within a small region that takes into account the locality of these approximations. Outline of such algorithms is presented below, where  $f(x)$  has to be minimized. Let us choose parameters  $\nu_1, \nu_2, \kappa_1, \kappa_2, \kappa_3$  so that  $0 < \nu_2 < \nu_1 < 1$ ,  $0 < \kappa_3 \leq \kappa_2 < 1 \leq \kappa_1$ :

Here is the schedule of an iteration :

1. Consider from the previous step the current Trust Region radius  $\rho_k$ , the current state  $x_k$ , and  $f^k(x) = f(x_k) + \nabla f(x_k)^T(x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k)$ .
2. Solve the optimization problem  $\tilde{x} = \operatorname{argmin} f^k(x)$  with  $\|x - x_k\| \leq \rho_k$ .
3. Compute  $f(\tilde{x})$  and define

$$\lambda = \frac{f(x_k) - f(\tilde{x})}{f^k(x_k) - f^k(\tilde{x})}$$

4. Update the trust region radius:

- If  $\nu_1 \leq \lambda$  set  $x_{k+1} = \tilde{x}$  and increase the trust region radius  $\rho_{k+1} = \kappa_1 \rho_k$ . Consider the next step  $k + 1$  and goto 1.
- If  $\nu_2 \leq \lambda \leq \nu_1$  set  $x_{k+1} = \tilde{x}$  and decrease the trust region radius  $\rho_{k+1} = \kappa_2 \rho_k$ . Consider the next step  $k + 1$  and goto 1.
- If  $\lambda \leq \nu_2$  set  $x_{k+1} = x_k$  and decrease the trust region radius  $\rho_{k+1} = \kappa_3 \rho_k$ . Consider the next step  $k + 1$  and goto 2.

When  $f$  is replaced by an optimal control system a POD-ROM approach is substituted to the successive quadratic functions, leading to the Trust Region Proper Orthogonal Decomposition method (TRPOD) introduced by [1]. Reduced model is updated at each iteration and embedded in a Trust-Region framework, in order to ensure the convergence of the algorithm regarding the way POD-ROM is built up.

Recall the initial discretized optimal control problem:

Minimize

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \sum_0^T \|\mathbf{Y} - \mathbf{Y}_d\|^2 + \frac{\delta}{2} \sum_0^T \|\gamma\|^2$$

subject to the discretized state constraint system defined for Navier Stokes equation, and written in a more compact form  $\mathbf{Y} = \mathbf{Y}_\gamma$ .

Recall that

$$\begin{aligned} F(\gamma) &= J(\gamma, \mathbf{Y}_\gamma) \\ F^p(\gamma) &= J(\gamma, \mathbf{Y}_\gamma^p) \end{aligned} \tag{32}$$

We follow the outline presented in [1] concerning TRPOD :

1. Consider  $\gamma_k$  the current rotation and  $\rho_k$  the trust region radius.
2. Compute  $\mathbf{Y}_{\gamma_k}$  and the corresponding snapshot set  $\{\mathbf{Y}_{\gamma_k}^1, \dots, \mathbf{Y}_{\gamma_k}^{nt}\}$
3. Compute POD-ROM and reduced cost function\*  $F^p$ .
4. Step Determination Algorithm: Solve the optimization problem

$$s_k = \operatorname{argmin} \quad F^p(\gamma_k + s),$$

with  $\|s\| \leq \rho_k$ .

5. Compute  $F(\gamma_k + s_k)$  and define

$$\lambda = \frac{F(\gamma_k) - F(\gamma_k + s_k)}{F^p(\gamma_k) - F^p(\gamma_k + s_k)}$$

6. Update the trust region radius:

- If  $\nu_1 \leq \lambda$  set  $\gamma_{k+1} = \gamma_k + s_k$  and increase the trust region radius  $\rho_{k+1} = \kappa_1 \rho_k$ . Consider the next step  $k + 1$  and goto 1.
- If  $\nu_2 \leq \lambda \leq \nu_1$  set  $\gamma_{k+1} = \gamma_k + s_k$  and decrease the trust region radius  $\rho_{k+1} = \kappa_2 \rho_k$ . Consider the next step  $k + 1$  and goto 1.
- If  $\lambda \leq \nu_2$  set  $\gamma_{k+1} = \gamma_k$  and decrease the trust region radius  $\rho_{k+1} = \kappa_3 \rho_k$ . Consider the next step  $k + 1$  and goto 4.

Fletcher-Reeves algorithm and Armijo Rule are both used for computing the solution at step 4:

1. Compute  $\nabla F^p(\gamma_k)$ , define  $s_k^0 = 0$  and  $h^0 = \nabla F^p(\gamma_k)$ .
2. for  $i = 1, \dots, n - 1$  do
3. Replace  $s_k^{i-1}$  by  $s_k^i$  where  $s_k^i = s_k^{i-1} + \lambda^{i-1} h^{i-1}$  and  $\lambda^{i-1}$  agrees with Armijo Rule.
4. If  $i < n$ , set

$$h^i = -\nabla F^p(\gamma_k + s_k^i) + \frac{\sum_0^T \|\nabla F^p(\gamma_k + s_k^i)\|^2}{\sum_0^T \|\nabla F^p(\gamma_k + s_k^{i-1})\|^2} h^{i-1}$$

---

\*The reduced cost function definition depends undirectly on the control  $\gamma_k$  that generated the POD basis

## 5.2 Coupled Reduced Order Models

Numerical results obtained in a TRPOD framework show problems of convergence when  $\gamma_k$  is close to an optimum. Prevailing errors that prevent the algorithm from fully converging are those introduced by POD-ROM approximation. In such case, Trust-Region radius sharply decreases, leading to unsuccessful iterations and an increase of computation time cost.

Coupling POD-ROMs would allow to overcome accuracy problems by taking into account test points used within an optimization loop. Moreover, the method is numerically built up with negligible extra cost.

Given 2 simulated velocity fields

$$(\gamma^1, \mathbf{Y1}), (\gamma^2, \mathbf{Y2})$$

One can extract 2 snapshots sets  $\{(\mathbf{Y1}^i)_{1 \leq i \leq nt}\}$  and  $\{(\mathbf{Y2}^i)_{1 \leq i \leq nt}\}$ . Reduced models can be obtained through 2 ways:

- Either by merging the 2 sets. A unique POD basis is extracted and thus a single POD-ROM  $\mathbf{Y}_\gamma^p$ .
- Either by leaving the 2 sets apart, generating 2 POD-ROMs  $\mathbf{Y}_\gamma^{p1}$  and  $\mathbf{Y}_\gamma^{p2}$ .

Numerically, we observe problems of accuracy by considering the single POD-ROM.

Recall that

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \sum_0^T \|\mathbf{Y} - \mathbf{Y}_d\|^2 + \frac{\delta}{2} \sum_0^T \|\gamma\|^2$$

Thus

$$J(\gamma, \mathbf{Y}) = \frac{1}{2} \sum_0^T \|\mathbf{Y} - \mathbf{Y}^p\|^2 + \frac{1}{2} \sum_0^T \|\mathbf{Y}^p - \mathbf{Y}_d\|^2 + \sum_0^T (\mathbf{Y} - \mathbf{Y}^p, \mathbf{Y}^p - \mathbf{Y}_d) + \frac{\delta}{2} \sum_0^T \|\gamma\|^2$$

Usually,  $\|\mathbf{Y} - \mathbf{Y}^p\|^2$  and  $(\mathbf{Y} - \mathbf{Y}^p, \mathbf{Y}^p - \mathbf{Y}_d)$  are expected to be negligible, according to Galerkin projection assumptions.

Assume that 2 POD-ROMs  $\gamma \rightarrow \mathbf{Y}^{p1}$  and  $\gamma \rightarrow \mathbf{Y}^{p2}$  were computed. Then,

$$J(\gamma, \mathbf{Y}) = \frac{1}{4} \sum_0^T \|\mathbf{Y}^{p1} - \mathbf{Y}^{p2}\|^2 + \frac{1}{2} (J(\gamma, \mathbf{Y}^{p1}) + J(\gamma, \mathbf{Y}^{p2})) + \frac{1}{2} \sum_0^T (\mathbf{Y} - \mathbf{Y}^{p1}, \mathbf{Y} - \mathbf{Y}^{p2}) + \frac{1}{2} \sum_0^T (\mathbf{Y} - \mathbf{Y}^{p1}, \mathbf{Y}^{p1} - \mathbf{Y}_d) + \frac{1}{2} \sum_0^T (\mathbf{Y} - \mathbf{Y}^{p2}, \mathbf{Y}^{p2} - \mathbf{Y}_d)$$

The two last terms are not taken into account. Therefore

$$J(\gamma, \mathbf{Y}) = \frac{1}{4} \sum_0^T \|\mathbf{Y}^{p1} - \mathbf{Y}^{p2}\|^2 + \frac{1}{2} (J(\gamma, \mathbf{Y}^{p1}) + J(\gamma, \mathbf{Y}^{p2})) + \frac{1}{2} \sum_0^T (\mathbf{Y} - \mathbf{Y}^{p1}, \mathbf{Y} - \mathbf{Y}^{p2})$$

Given  $\gamma$ ,  $Y(\gamma)$  is replaced by its estimation

$$\tilde{\mathbf{Y}}(\gamma) = \bar{\theta}\mathbf{Y}^{p1}(\gamma) + (1 - \bar{\theta})\mathbf{Y}^{p2}(\gamma) \quad (33)$$

where  $\bar{\theta}$  is solution to

$$\begin{cases} \text{Min } R(\theta, \mathbf{Z}) = \sum_0^T \|\theta\mathbf{Y}^{p1}(\gamma) + (1 - \theta)\mathbf{Y}^{p2}(\gamma) - \mathbf{Z}\|^2 \\ \theta \in [0, 1] \\ \mathbf{Z} \in \text{span}(\mathbf{Y1}, \mathbf{Y2}) \end{cases} \quad (34)$$

This optimal parameter  $\theta$  can be understood as a will to choose the reduced model that deviates as less as possible to the velocity fields that generated the two POD-ROMS.

Finally, the reduced cost function  $F^p$  is

$$\frac{1}{2} \left\{ \frac{1}{2} \sum_0^T \|\mathbf{Y}^{p1} - \mathbf{Y}^{p2}\|^2 + (F^{p1} + F^{p2}) + \sum_0^T \left( \tilde{\mathbf{Y}} - \mathbf{Y}^{p1}, \tilde{\mathbf{Y}} - \mathbf{Y}^{p2} \right) \right\}$$

### 5.3 Multiobjective optimization

Around a local minimizer of  $F$ , problems of convergence is numerically observed. [5] have introduced a POD-ROM model of the adjoint equation in order to have a better approximation of the cost function and its local behaviour. Instead, we propose to simply use  $\nabla F$  inside an optimization loop of TRPOD in accordance with the following remarks :

- The gradient norm of  $F$  is high when far from a local minimizer. It's possible to deviate from the steepest descent direction to obtain a minimum inside the trust-region.
- The gradient norm of  $F$  is low when close to a local minimizer. It's preferable to search a minimizer along the steepest descent direction inside the trust region.

A basic approach would consist in introducing the gradient information as a direction constraint for searching a minimum inside an optimization loop of TRPOD algorithm. The resulting problem would be

$$\begin{cases} s_k = \text{argmin}_{\|s\| \leq \rho_k} F^p(\gamma_k + s) \\ \nabla F(\gamma_k) \cdot s \leq 0 \end{cases}$$

However the gradient norm information is not available and the previous strategy cannot be implemented straightforward. Let's introduce the first order approximation of  $F$ , denoted by  $F^2$  which is defined by

$$F^2(\gamma) = F(\gamma_k) + \nabla F_{\gamma_k} \cdot (\gamma - \gamma_k)$$

Considering  $F^1(s) = F^p(\gamma_k + s)$  as a low order approximation of  $F$  around  $\gamma_k$ , we say that  $s$  is admissible if

$$\begin{cases} F^1(s) - F^1(0) \leq 0 \\ F^2(s) - F^2(0) \leq 0 \end{cases}$$

Define  $\mathbf{F} = (F^1, F^2)$ .  $s_1$  is said to be better than  $s_2$  if  $\mathbf{F}(s_1) \leq \mathbf{F}(s_2)$ , that is

$$\begin{cases} F^1(s_1) - F^1(s_2) \leq 0 \\ F^2(s_1) - F^2(s_2) \leq 0 \end{cases}$$

Instead of minimizing  $F^1$ , we seek for a Pareto optimal point  $s_k$  of  $\mathbf{F}$ , according to the next definition.

**Definition.**

A point  $\tilde{s}$  is said to be Pareto optimal if and only if there is no  $s$  such that  $\mathbf{F}(s) \leq \mathbf{F}(\tilde{s})$  with at least one strict inequality:  $F^1(s) < F^1(\tilde{s})$  or  $F^2(s) < F^2(\tilde{s})$ .

Pareto front is the set  $K$  of Pareto optimal points.

In order to reach the Pareto front, an admissible descent direction has to be redefined for multiobjective optimization.

**Definition.**

Define  $\alpha \in ]0, 1[$ .  $d$  is said to be an admissible descent direction or agrees with Armijo rule if

$$\exists t > 0, \mathbf{F}(td) \leq \mathbf{F}(0) + \alpha t J\mathbf{F}(0)d$$

where  $J\mathbf{F}(0)$  is the Jacobian of  $\mathbf{F}$  at 0.

In the scalar case, the steepest descent is admissible. In the vector case, steepest descent needs to be redefined.

[6] suggested to use the following definition :

**Definition.**

Steepest descent of  $\mathbf{F}$  is defined by normalized vector  $v$  solution to optimization problem

$$\text{Min}_v \quad \text{Max}_{i=1,2} (v, \nabla F^i) + \frac{1}{2} \|v\|^2$$

and agrees with Armijo Rule.

We introduce in the sequel  $\nabla_2 \mathbf{F} \equiv -v$ .

This steepest direction agrees with the previous remarks. Assume that  $(\nabla F^2, \nabla F^1) \geq 0$ :

- When  $\|\nabla F^2\|$  tends to 0 (for instance close to an optimum),  $\nabla_2 \mathbf{F} = \nabla F^2$ . The steepest descent of  $F$  is selected. Therefore optimization using POD-ROM is not relevant.
- When  $\|\nabla F^2\|$  is sufficiently large,  $\nabla_2 \mathbf{F} = \nabla F^1$ . The steepest direction  $\nabla F^2$  is not a key direction for optimization. Therefore optimization using POD-ROM is relevant.

## 5.4 Algorithm

The final algorithm uses the two improvements described in the previous section. Instead of finding point  $\gamma_{k+1}$  that minimizes  $F^p$ , a Pareto optimal point of  $\mathbf{F}$  is computed inside an optimization loop. Moreover, depending on the correlation criteria value introduced in TRPOD, either a single POD-ROM model or a coupled one is considered. Recall that minimizing a vector cost function consists in finding an optimal Pareto point.

First define the following useful parameters :

- Correlation criteriae:  $0 < \nu_2 < \nu_1 < 1$ .
- Trust-Region radius parameters:  $0\kappa_3 \leq \kappa_2 < 1 \leq \kappa_1$ .
- Coupling boolean  $coupling \in \{0, 1\}$ .

1. Compute  $\mathbf{Y}_{\gamma_k}$  from  $\gamma_k$  and  $\nabla F(\gamma_k)$ . Define POD-ROM  $Y_{\gamma}^{p1}$  and  $F^2(s) = F(0) + \nabla F_{\gamma_k} \cdot s$ .

2. Consider  $\mathbf{F}$ :

- If  $coupling = 0$ ,  $F^1(s) = F^{p1}(\gamma_k + s)$  where  $F^{p1}(\gamma) = J(\gamma, Y_{\gamma}^{p1})$ .
- If  $coupling = 1$ ,  $F^1(s) = F^p(\gamma_k + s)$  where

$$\begin{aligned} F^p(\gamma) &= \frac{1}{2} \left\{ \frac{1}{2} \sum_0^T \|\mathbf{Y}_{\gamma}^{p1} - \mathbf{Y}_{\gamma}^{p2}\|^2 + (F^{p1}(\gamma) + F^{p2}(\gamma)) + \sum_0^T \left( \tilde{\mathbf{Y}} - \mathbf{Y}^{p1}, \tilde{\mathbf{Y}} - \mathbf{Y}^{p2} \right) \right\} \\ F^{p1}(\gamma) &= J(\gamma, \mathbf{Y}_{\gamma}^{p1}) \\ F^{p2}(\gamma) &= J(\gamma, \mathbf{Y}_{\gamma}^{p2}) \end{aligned} \quad (35)$$

and  $\tilde{\mathbf{Y}}$  computed according to (33).

3. Solve optimization problem :

$$s_k = \operatorname{argmin}_{\|s\| \leq \delta_k} \mathbf{F}(s)$$

where  $\mathbf{F} = (F^1, F^2)$ .

4. Compute  $F(\gamma_k + s_k)$  and

$$\lambda = \frac{F(\gamma_k) - F(\gamma_k + s_k)}{F^1(0) - F^1(s_k)}$$

5. Update Trust-Region radius and POD-ROM :

- If  $\nu_1 \leq \lambda$ ,  $\gamma_{k+1} = \gamma_k + s_k$  and  $\delta_{k+1} = \kappa_3 \delta_k$ . Set  $Y^{p2} = Y^{p1}$ ,  $couplage = 1$ . Consider the next step  $k + 1$  and goto 1.
- If  $\nu_2 \leq \lambda < \nu_1$ ,  $\gamma_{k+1} = \gamma_k + s_k$  and  $\delta_{k+1} = \kappa_2 \delta_k$ . Set  $couplage = 0$ . Consider the next step  $k + 1$  and goto 1.
- If  $\lambda < \nu_2$ ,  $\gamma_{k+1} = \gamma_k$  and  $\delta_{k+1} = \kappa_1 \delta_k$ . Compute POD-ROM  $\mathbf{Y}_{\gamma}^{p2}$  from  $\mathbf{Y}_{\gamma_k + s_k}$  and set  $couplage = 1$ . Consider the next step  $k + 1$  and goto 2.

Fletcher-Reeves algorithm and Armijo Rule in vector case are both used for computing the solution at step 3:

1. Compute  $\nabla_2 \mathbf{F}(0)$ , define  $s_k^0 = 0$  and  $h^0 = \nabla_2 \mathbf{F}(0)$ .
2. for  $i = 1, \dots, n - 1$  do
3. Replace  $s_k^{i-1}$  by  $s_k^i$  where  $s_k^i = s_k^{i-1} + \lambda^{i-1} h^{i-1}$  and  $\lambda^{i-1}$  agrees with Armijo Rule.
4. If  $i < n$ , set

$$h^i = -\nabla_2 \mathbf{F}(s_k^i) + \frac{\sum_0^T \|\nabla_2 \mathbf{F}(s_k^i)\|^2}{\sum_0^T \|\nabla_2 \mathbf{F}(s_k^{i-1})\|^2} h^{i-1}$$

## 6 Numerical Results

### 6.1 Configuration

The configuration presented below is used for numerical computations. The cylinder has diameter  $D$  equal to 1, and is 10 units far from the left (inflow), upper and lower boundaries of domain  $\Omega$ .

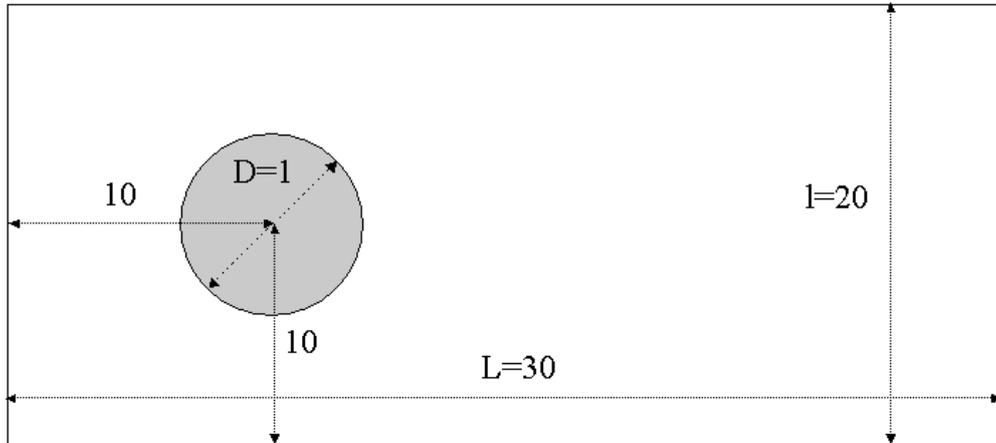


Figure 3: Configuration

We recall that :

- A Dirichlet boundary condition of the form  $\mathbf{Y} = \mathbf{Y}_\infty$  is enforced on the left, upper and lower boundaries.
- An outflow boundary condition  $\partial_n \mathbf{Y} - p\mathbf{n} = 0$  is used at the right boundary.
- On the cylinder boundary, we have  $\mathbf{Y} = \gamma \mathbf{t}$

An unstructured mesh of size 7000 is built up, with refinements around the cylinder and in its wake.

### 6.2 Validation of the Solver

The correctness of optimization results sharply depends on both configuration and solver accuracy. Therefore, as a preliminary to optimization numerical results, a validation of our solver is made for several Reynolds numbers ranging from 10 to 200.

For Reynolds number values below 40, a symmetric steady flow is observed. A recirculating zone appears in the wake of the cylinder, with length positively correlated to  $Re$ , as shown in Figure 4

For  $Re$  above a critical value around 40, the control free flow becomes unsteady, although unconditionally stable. As shown in fig. 5, the symmetry is broken and a well-known Von-Karman vortex street appears behind the cylinder, with periodicity defined by  $S_t$ .

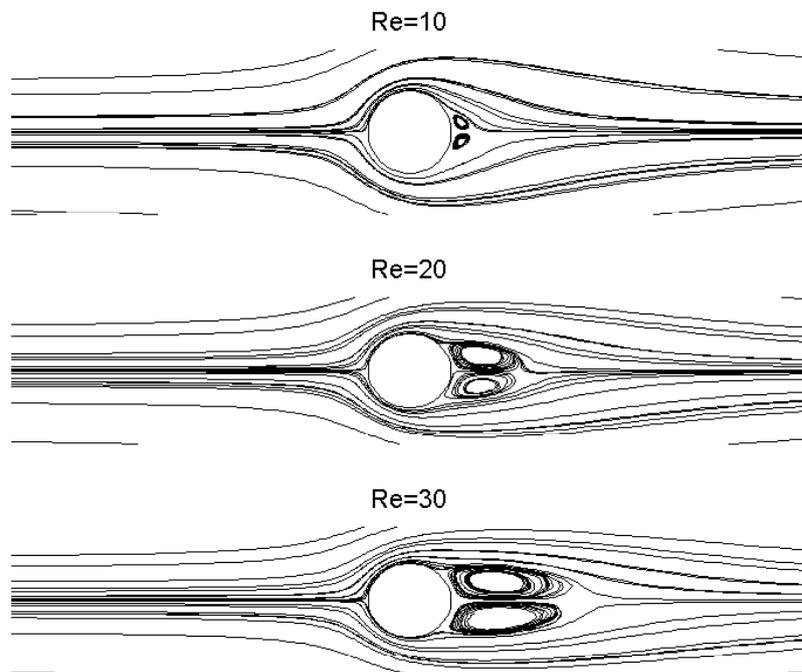


Figure 4: Control free flow



Figure 5: Flow lines for low Reynolds number

Numerically, when taking as initial value the corresponding unstable steady flow, a sharp increase of the drag coefficient is first observed during the transient regime as shown in fig. 6.

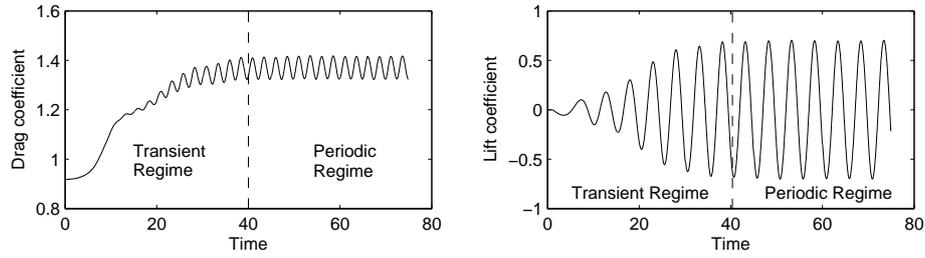


Figure 6: Evolution of Drag and Lift coefficients

At a stage a stable periodic regime appears for  $C_d$  and  $C_l$  with their values stabilizing around a mean one constant in time. When analysing the power spectrum of  $C_d$  and  $C_l$ , their oscillations are governed by a predominant fundamental frequency, namely  $f_s$  for  $C_l$  through which  $S_t$  is easily computed. Figure 7 shows the behaviour of  $C_d$  and  $C_l$  for  $Re = 200$  during the periodic regime.  $C_d$  and  $C_l$  oscillate around their mean value, respectively  $\bar{C}_d = 1.3682$  and  $\bar{C}_l = 0.0047$  where

$$\bar{C}_d \mathbf{e}_x + \bar{C}_l \mathbf{e}_y = \frac{1}{T} \int_0^T \int_{\Gamma_c} \left\{ \frac{P - P_\infty}{\frac{1}{2} \rho \|\mathbf{Y}_\infty\|_2^2} - \frac{2}{Re} \frac{\partial \mathbf{Y}}{\partial n} \right\} d\Gamma dt$$

The power spectrum of  $C_l$  leads to  $S_t = 0.2000$ .

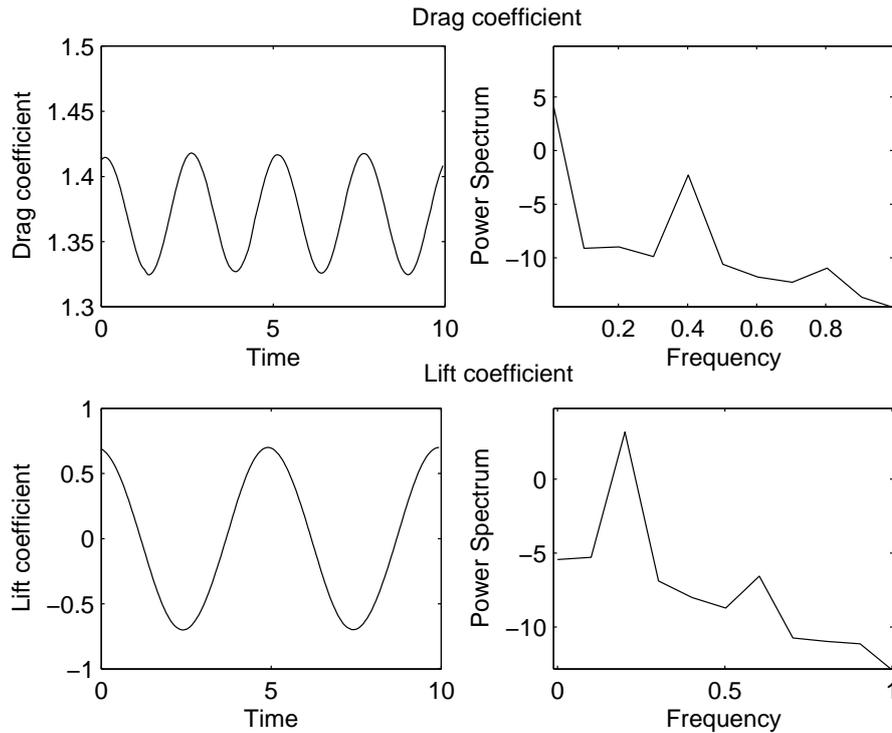


Figure 7: Control free flow and controlled one

As presented before, steady flows become unstable for  $Re > 40$ . They can be computed

as stationary solution of the Navier-Stokes equation. Nevertheless characteristics of such flows are worth being mentioned, as they show :

- A low drag coefficient  $C_d$  compared to the unsteady flow computed for the same Reynolds number (cf. fig. 8)
- A desired profile with low vorticity values.

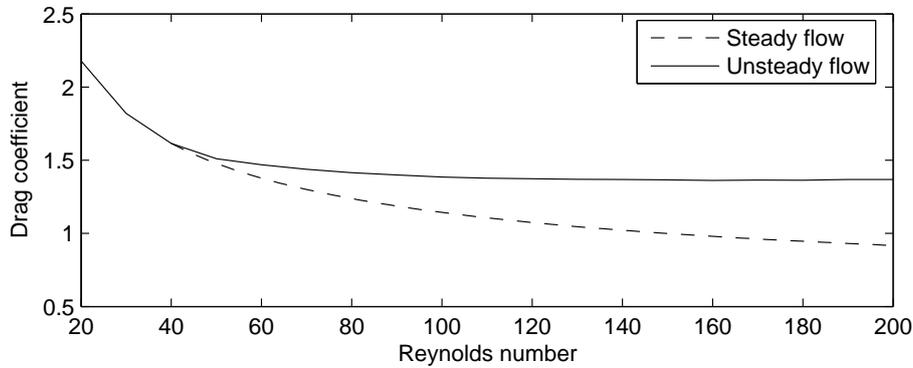


Figure 8: Drag coefficient for steady and unsteady flows

Datas numerically computed by several authors are summarized in Tables 1, 2 and in Figure 9 for various Reynolds number values, ranging from 20 to 200.

Table 1: Drag coefficient values

$Re$	Present Work	Bergmann et <i>al.</i>	He et <i>al.</i>	Braza et <i>al.</i>	Henderson
20	2.1779	2.2500	2.0064	2.1900	2.0587
40	1.6154	1.6800	1.5047	1.5800	1.5445
60	1.4695	-	1.3859	1.3500	1.4151
80	1.4150	-	1.3489	-	1.3727
100	1.3846	1.4100	1.3528	1.3600	1.3500
200	1.3682	1.3900	1.3560	1.3900	1.3412

Table 2: Drag coefficient values

$Re$	Present Work	Bergmann et <i>al.</i>	He et <i>al.</i>	Braza et <i>al.</i>	Henderson
60	0.1400	-	0.1353	-	0.1379
80	0.1570	-	0.1526	-	0.1547
100	0.1670	0.1670	0.1670	0.1600	0.1664
200	0.2000	0.1990	0.1978	0.2000	0.1971

For low values of Reynolds number ( $Re < 100$ ), Strouhal number computed through our method seems to be slightly overpredicted. However, for our typical configuration where  $Re = 200$ , good numerical values are computed, close to the ones presented above.

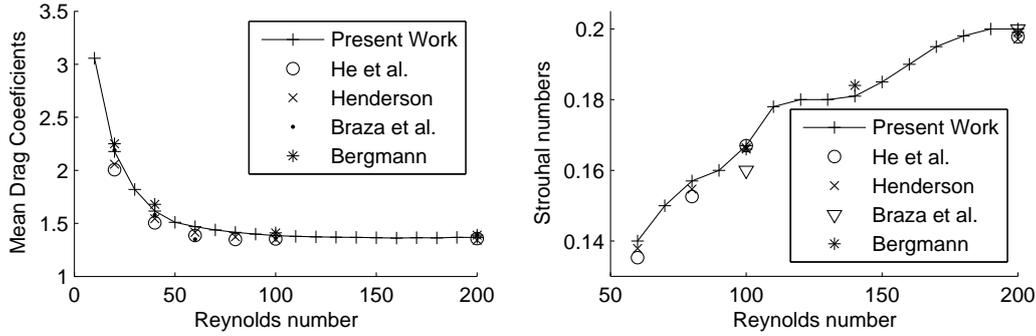


Figure 9: Comparison between several solvers

## 6.3 Results obtained with BFGS Algorithm (He et al.)

### 6.3.1 Presentation

In order to validate the results obtained by our algorithm, BFGS method is used as reference for result comparisons. The mesh size of our configuration is low enough to use a simple implementation of BFGS algorithm instead of a limited memory one suggested by [16], when facing large scale optimization.

Here is the short presentation of BFGS method that is given in [11] for BFGS method description, where  $f$  is defined on  $\mathbb{R}^N$  and  $g(x) = \nabla f(x)$ .

- **Initialization.** Choose  $0 < \alpha < \frac{1}{2}$ ,  $\alpha < \beta < 1$ ,  $x^0$ ,  $H^0 \in \mathcal{L}(\mathbb{R}^N, \mathbb{R}^N)$  and  $g^0 = \nabla f(0)$ .
- **Optimization loop.**

1. Compute  $d^k = -H^k g^k$ ,  $x^{k+1} = x^k + \rho^k d^k$  where  $\rho^k$  is chosen according to the following conditions :

$$\begin{cases} f(x^k + \rho^k d^k) \leq f(x^k) + \alpha \rho^k (g^k, d^k) \\ (g(x^k + \rho^k d^k), d^k) \geq \beta (g^k, d^k) \end{cases}$$

2. Update  $H^k$  thanks to the following formula

$$\begin{aligned} s^k &= x^{k+1} - x^k \\ y^k &= g^{k+1} - g^k \\ H^{k+1} &= H^k + \frac{(s^k - H^k y^k) \otimes s^k + s^k \otimes (s^k - H^k y^k)}{(y^k, s^k)} - \frac{(s^k - H^k y^k, y^k)}{(y^k, s^k)^2} s^k \otimes s^k \end{aligned}$$

where  $u \otimes v$  is the usual tensor product.

In our case, we have

$$x \in \{(A, f, \phi), A \in \mathbb{R}^+, f \in \mathbb{R}^+, \phi \in [0, 2\pi]\} \subset \mathbb{R}^3$$

and

$$f(x) \equiv F(\gamma_x) = \frac{1}{2} \int_0^T \|\mathbf{Y} \gamma_x - \mathbf{Y}_d\|^2 dt + \frac{10^{-3}}{2} \int_0^T \gamma_x^2(t) dt$$

where

$$\gamma_x = x_1 \sin(2\pi x_2 t + x_3)$$

We recall that  $\mathbf{Y}_d$  is an unforced steady flow computed for  $Re_d \leq Re$ .

### 6.3.2 Computation of the desired profile $\mathbf{Y}_d$

For clarity reasons, let's denote by

- $NSC(\mathbf{Y}, P, \gamma) = 0$  the compact form of state equation.
- $NSS(\mathbf{Y}_d, P_d, Re_d) = 0$  the stationary state equation satisfied by  $\mathbf{Y}_d$ .
- $J(\mathbf{Y}, \gamma, \mathbf{Y}_d, Re_d) = \frac{1}{2} \|\mathbf{Y} - \mathbf{Y}_d\|^2 + \frac{10^{-3}}{2} \|\gamma\|^2$ .

Numerical results obtained for flow tracking cost function type is quite sensitive to  $\mathbf{Y}_d$  (and the corresponding  $Re_d$ ). Let's consider the following cost function :

$$Re_d \rightarrow \begin{cases} \text{Min}_{\gamma} J(\mathbf{Y}, \gamma, \mathbf{Y}_d, Re_d) \\ NSC(\mathbf{Y}, P, \gamma) = 0 \\ NSS(\mathbf{Y}_d, P_d, Re_d) = 0 \end{cases}$$

Table 3 shows that optimization results differ from one typical value of  $Re_d$  to another.

Table 3: Optimization results for typical values of  $Re_d$

$Re_d$	$A$	$f$	Cost function Value	$C_d$
5	2.49	0.58	79.56	1.03
30	3.00	0.73	66.07	0.99
200	3.45	0.72	122.79	1.01

The resulting drag coefficient value is the lowest for  $Re_d = 30$ . Indeed optimal parameters ( $A, f$ ) for  $\gamma = A \sin(2\pi ft + \phi)$  are the closest to the ones found in [11]. One can finally remarks that the cost functional  $\text{Min}_{\gamma} J(\mathbf{Y}, \gamma, \mathbf{Y}_d, Re_d)$  with respect to constraints  $NSC(\mathbf{Y}, P, \gamma) = 0$  and  $NSS(\mathbf{Y}_d, P_d, Re_d) = 0$  is minimized for such value of  $Re_d$ .

By considering the following optimization problem :

$$\begin{cases} \text{Min}_{\gamma, Re_d} J(\mathbf{Y}, \gamma, \mathbf{Y}_d, Re_d) \\ NSC(\mathbf{Y}, P, \gamma) = 0 \\ NSS(\mathbf{Y}_d, P_d, Re_d) = 0 \end{cases}$$

Minimum is reached for

$$(Re_d, A, f) = (30, 3.00, 0.73)$$

and agrees with previous results. The corresponding cost function value is  $J = 66.07$ . On can remark that the appropriate profile for  $\mathbf{Y}_d$  seems to be the solution of the previous optimization problem. However, such problem is hard to be implemented due to the increase of the number of high dimension unknowns to be optimized ( $(\mathbf{Y}, \mathbf{Y}_d)$ ). Therefore it's necessary to compute  $\mathbf{Y}_d$  in a different way. One can remark that variations of

$J(\mathbf{Y}, \gamma, \mathbf{Y}_d, Re_d)$  is more sensitive to  $(Re_d, \mathbf{Y}_d)$  than  $(\gamma, \mathbf{Y})$ . That is, one can expect to retrieve optimal value of  $(Re_d, \mathbf{Y}_d)$  if  $(\gamma, \mathbf{Y})$  remains fixed. Numerically,  $\gamma$  was chosen to be equal to 0 while  $\mathbf{Y}$  is simply the control free flow at  $Re = 200$  (denoted by  $\mathbf{Y}_0$ ). In a mathematical formulation, it consists in finding  $Re_d$  solution to :

$$\begin{cases} \text{Min}_{Re_d} J(\mathbf{Y}_0, 0, \mathbf{Y}_d, Re_d) \\ NSS(\mathbf{Y}_d, P_d, Re_d) = 0 \end{cases}$$

Figure 10 shows behaviour of  $Re_d \rightarrow J(\mathbf{Y}_0, 0, \mathbf{Y}_d, Re_d)$  with respect to constraint

$$NSS(\mathbf{Y}_d, P_d, Re_d) = 0$$

The previous assumption is verified by remarking that optimal value of  $Re_d$  is approximately equal to 30.

The closest  $\mathbf{Y}_d$  profile to  $\mathbf{Y}_0$  in  $L^2$  sense leads to the best results for our optimal control problem in terms of drag reduction. This result seems to be all the more paradoxal that  $C_d$  value for stationary flow  $\mathbf{Y}_d$  is greater than the one obtained for  $\mathbf{Y}$  at  $Re = 200$  when no control is enforced at the cylinder boundary (cf. Figure 8). Let's recall that we don't aim at reducing directly  $C_d$  through minimization of  $J$ . Instead, the objective is to reduce vortex shedding by reaching an appropriate flow profile  $\mathbf{Y}_d$ . Undirectly, we expect and observe that reducing vortex shedding leads to drag reduction and this independently to  $C_d$  value of  $\mathbf{Y}_d$ .

The fact that the best profile  $\mathbf{Y}_d$  is computed for  $Re_d = 30$  seems to be linked with the length of its recirculating zone. One can observe numerically that this length is close to the vortex one.

In the sequel, we use  $\mathbf{Y}_d$  computed at  $Re_d = 30$ .

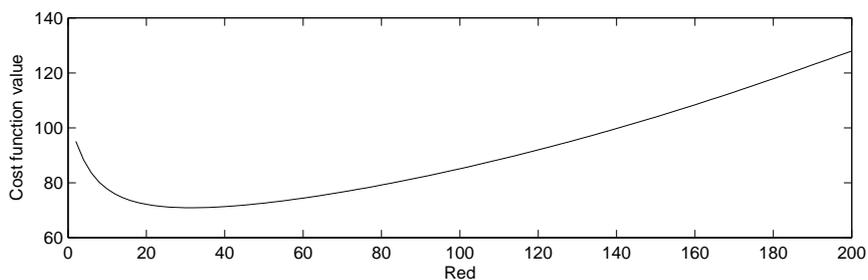


Figure 10: Values of  $G(Re_d)$

## 6.4 Convergence of our algorithm

Our algorithm was initialized for several values of  $\gamma$ . In most of the case, it converged to the following sub-optimal parameter :

$$(\hat{A}, \hat{f}, \hat{\phi}) = (2.85, 0.75, 0.78)$$

that is close to the one obtained with BFGS for  $Re_d = 30$ .

Behaviour of the algorithm is quite independent to initialization. Table 4 shows its behaviour for a typical initial condition.

As a first step, a sharp decrease of cost functional  $F$  is observed at the upper part of Table 4. Indeed, our coupled POD-ROM model is in most of the case a good prediction of flow behaviour as  $\rho > 0.5$ . For comparison purposes, evolution of TRPOD is shown in Table 5. A slower decrease is observed for TRPOD at this step. In order to know whether our coupled POD-ROMS models is involved in this sharp reduction or not we compare the fiability of both models within a trust region that is centered at  $\gamma$ , where  $k$  is selected from the first step. A random exploration for the control  $\gamma$  is made within this trust region.  $\rho$  can now be considered as a random variable (as a function of  $\gamma$ ), where we recall that

$$\rho = \frac{F(\gamma_k) - F(\gamma)}{F^p(\gamma_k) - F^p(\gamma)}$$

and is the fiability criteria of the reduced cost function  $F^p$  at  $\gamma$ .

The probability distribution of  $\tilde{\rho} = \min(\rho, \frac{1}{\rho})$  is shown at Table 6.

Relevancy of both reduced order models are comparable at this step of our algorithm. Therefore, the significant decrease of  $F$  is due to the multiobjective part of our reduced cost function model, where the gradient information was included.

During the second step of our algorithm, a decrease of trust region radius is observed when close to an optimum, corresponding to a slow decrease of  $F$ . In comparison, TRPOD algorithm fails in fully converging as observed in Table 5. One can remark through Table 6 that  $\tilde{\rho}$  probability density has values concentrated around 0 for simple POD-ROM model. Thus, it is not relevant enough to be a good prediction of flow behaviour. The same remarks could be applied to the coupled POD-ROMs when facing for the first time an unsuccessful test within an iteration loop (Coupled model number 1 in Table 6). However in such case the coupling is updated, leading to another prediction (Coupled model number 2). The probability density switched from values around 0 to values above 0.5, which means that the new prediction model is good enough to go on the optimization process as shown in Table 4.

Iteration number	$n_p$	function cost value	$\rho_k$
Step 1			
1	88	69.64	1.04
2	87	68.55	0.75
3	89	67.47	0.23
4	92	67.21	0.62
5	91	67.02	0.72
6	93	66.78	0.80
7	95	66.51	0.70
8	95	66.31	0.56
Step 2			
9	98	66.23	0.23
12	118	66.20	0.52
13	121	66.15	0.60
15	147	66.10	0.50

Table 4: Convergence results for our Algorithm

Iteration number	$n_p$	function cost value	$\rho_k$
Phase 1			
1	90	69.64	0.18
2	90	69.57	0.15
3	90	69.52	0.26
4	90	69.46	0.52
5	91	69.44	0.50
Phase 2			
6	91	69.44	-1.7
...	...	...	...

Table 5: Convergence results for TRPOD

Modèle POD	$p(\tilde{\rho} \leq 0)$	$p(0 < \tilde{\rho} \leq 0.5)$	$p(0.5 < \tilde{\rho})$
Phase 1			
simple	0.09	0.32	0.59
couplé	0.09	0.29	0.62
Phase 2			
simple	0.68	0.19	0.13
couplé 1	0.59	0.23	0.18
couplé 2	0.15	0.31	0.54

Table 6: Fiability of POD-ROMs models

When the optimal control is applied to the cylinder boundary, flow dynamics follows 2 regimes.

A transient regime first appears in a time window of length 30. The reduction of  $C_d$  is significant (28%) regarding the original one of the periodic control free case (cf. Figure 13).

A periodic regime then follows where  $C_d$  oscillates around its mean value 0.99. Vortex shedding almost disappears past the cylinder with residuals observed at the limits of the wake (see Fig. 11). The spectral analysis of  $C_l$  in Figure 12 shows its predominant frequency is identified with the cylinder rotation one. In other way, flow dynamics is totally managed by this rotation.

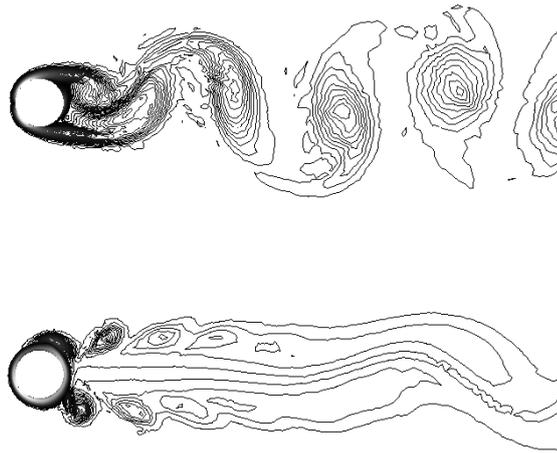


Figure 11: Control free flow and controlled one

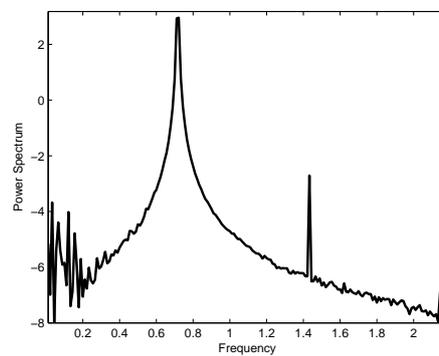
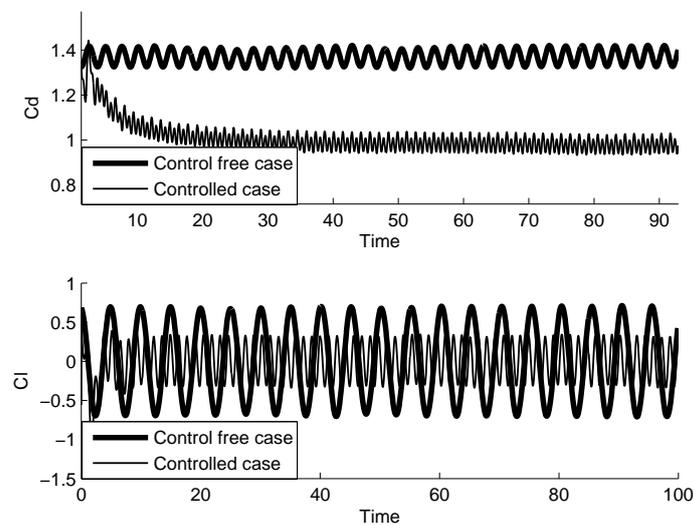
Figure 12: Spectral Analysis of  $C_l$ 

Figure 13: Drag and Lift coefficients evolution

## 7 Conclusions

Optimal Control Theory provides a significant framework for designing a relevant control that reduces vortex sheddings in a cylinder wake. It becomes all the more attractive that the corresponding optimization methods are less computationally demanding. In this way our improvement of TRPOD algorithm is promising. When it is applied to the cylinder wake flow control problem, we observe both time cost reduction and convergence towards well known results. Through the use of Coupled Low-Order Prediction Models and Multiobjective Optimization, accuracy and convergence difficulties encountered when prediction models are used within an optimization loop are overcome. However our numerical configuration is quite restrictive as we use a parameterized control applied to a circular cylinder in laminar regime. Thus the next steps would consist in validating our algorithm under smoothed conditions, in particular when Reynolds number increases, leading to tridimensional effects concerning flow behaviour.

## References

- [1] E. Arian, M. Fahl, and E. Sachs. Trust region proper orthogonal decomposition for optimal flow control. *NASA/CR-2000-210124*, 2000.
- [2] M. Bergmann, L. Cordier, and J.-P. Brancher. Control of the cylinder wake in the laminar regime by trust-region methods and pod reduced order models. In *IEEE Decision and Control, ECC*, 12 2005.
- [3] G. Berkooz, P. Holmes, and J.L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid. Mech.*, 25:539–575, 1993.
- [4] A.J. Chorin. The numerical solution of the navier-stokes equations. *Math. Comp.*, 22:745–762, 1968.
- [5] M. Fahl and E.W. Sachs. Reduced order modelling approaches to pde-constrained optimization based on proper orthogonal decomposition. In *Springer-Verlag, Large-Scale PDE-Constrained Optimization, Vol. 30*, 2003.
- [6] J. Fliege and B.F. Svaiter. Steepest descent methods for multicriteria optimization. *Math. Meth. of Operations Research*, 51(3):479–494, 2000.
- [7] W.R. Graham, J. Peraire, and K.Y. Tang. Optimal control of vortex shedding using low order models. *Int. J. for Num. Meth. in Eng.*, 44(7):845–990, 1999.
- [8] P.M. Gresho and S. Chan. On the theory of semi-implicit projectin methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix, part I : Theory. *Int. J. numer. Meth. Fluids*, 11:587–620, 1990.
- [9] J.-L. Guermond and L. Quartapelle. Calculation of incompressible viscous flows by an unconditionally stable projection fem. *J. Comput. Phys.*, 132:12–33, 1997.
- [10] M.D. Gunzburger, L. Hou, and T.P. Svobodny. Boundary velocity control of incompressible flow with an application to viscous drag reduction. *SIAM J. Control Optim.*, 30(1):167–181, 1992.

- 
- [11] J.-W. He, R. Glowinski, R. Metcalfe, A. Nordlander, and J. Periaux. Active control and drag optimization for flow past a circular cylinder. *J. Comput. Phys.*, 163(35):83–117, 2000.
- [12] C. Homescu, I.M. Navon, and Z. Li. Suppression of vortex shedding of flow around a circular cylinder using optimal control. *IJNMF*, 38:43–69, 2001.
- [13] G. Jin and M. Braza. A nonreflecting outlet boundary condition for incompressible unstead navier-stokes calculations. *J. Comp. Phys.*, 107(2):239–253, 1993.
- [14] J. Van Kan. A second-order pressure correction scheme for viscous incompressible flow. *SIAM J. Sci. Stat. Comput.*, 7:870–891, 1986.
- [15] J. Kim and P. Moin. Application of a fractinal-step method to incompressible navier-stokes equations. *J. Comp. Phys.*, 59:308–325, 1985.
- [16] D. C. Liu and J. Nocedal. On the limited memory bfgs method for large optimization. *Math. Prog.*, 45(503), 1989.
- [17] B. Protas and A. Styczek. Drag force in the open-loop control of the cylinder wake in the laminar regime. *Phys. Fluids*, 14(2):810–826, 2002.
- [18] B. Protas and A. Styczek. Optimal rotary control of the cylinder wake in the laminar regime. *Phys. Fluids*, 14(7):2073–2087, 2002.
- [19] R. Temam. *Navier-Stokes equations and nonlineat functional analysis*. SIAM, 1995.



**This page has been purposely left blank**