Fluid Flow Control: a Vision-Based Approach

Roméo Tatsambon Fomena, Christophe Collewet

HAL Id: hal-00701080
https://hal.inria.fr/hal-00701080

Submitted on 24 May 2012

Fluid Flow Control: a Vision-Based Approach

Romeo Tatsambon Fomena and Christophe Collewet

Abstract—**This paper proposes a new approach to control a flow. Controlling a flow consists either to change its state to another state or to maintain its current state whatever external disturbances. Here the control of the laminar plane Poiseuille flow is considered. To estimate the state of this flow, existing control methods rely on a set of limited wall shear stress measurements. These existing methods suffer from limited observations, from noisy measurements and from the initialization involved in the observer required to estimate the flow state. To deal with these issues, this paper proposes a vision-based control approach. More precisely, by visualizing a fluid flow, dense flow velocity maps can be computed via optical flow techniques and subsequently used to build an observer-free closed-loop control law. This approach is formally proven to be of great improvements for the control of this flow in comparison with existing control approaches.**

I. INTRODUCTION

A turbulent flow presents better mixing properties than a laminar flow. A significant part of the work carried out in the field of flow control has been dedicated to the control of the transition from laminar to turbulent states. Delaying, accelerating or modifying this transition can be of great environmental and economical interests for industrial applications. For instance, Airbus expects in 2020 to decrease by 50% the CO2 emissions, a large part of this decrease being expected from flow control by diminishing the fuel consumption of their aircrafts through drag reduction [1]. In contrast, in other application domains such as industrial chemistry, turbulence phenomena are encouraged to increase heat exchange, to improve the mixing of chemical components and to enhance chemical reactions.

Flow control can be achieved in two different ways: passive or active control. Passive control provides a permanent action on the system to control. Most often it consists in optimizing shapes or in choosing suitable materials such as riblets [2], [3], porous media [4] or hairy coatings [5]. Conversely, in active control an external energy is required to act on the system, like for example techniques based on blowing and suction [6] or based on a cylinder of rotation [7], [8]. This type of approach can be seen as an optimal problem where one has to apply an optimal control law based on a certain cost (minimization of the drag, minimization of the actuators power, etc.) [9]. However, very often, open-loop control strategies [10], [11] or even most often forcing strategies [7], [12], [13] are used. These strategies, contrary to a closed-loop control, necessitate an accurate knowledge of the flow [14], [13], and are not robust to the variations of unmodelled parameters of the

R. Tatsambon Fomena is with Cemagref / INRIA Rennes Bretagne-Atlantique, Fluminance team, Rennes 35042, France `romeo.tatsambon@cemagref.fr`

C. Collewet is with Cemagref / INRIA Rennes Bretagne-Atlantique, Fluminance team, Rennes 35042, France `christophe.collewet@inria.fr`

system. As a matter of fact, designing a closed-loop control law requires the use of sensors that can be at the same time non-intrusive, accurate and adapted to the time and space scales of the phenomenon under monitoring. Unfortunately, non-intrusive sensors are hardly available in the real context of control applications. The most commonly used measurement, obtained from Micro Electro Mechanical Systems (MEMS), is the shear stress at a limited set of measurement points on the wall [15], [16], [17]. A literature review on current and future developments, experimental use, and limitations of MEMS based shear stress sensors can be found in [18], [19].

Concerning Poiseuille flow, because of the existence of potentially unobservable high transient flow modes due to these limited observations, a simple output feedback control law cannot be used [15]. That is why current Poiseuille flow control approaches focus on the use of full state information. The reconstruction of the full state vector using limited wall shear stress measurements requires the use of an observer [16], [20], [21]. However an observer is sensitive to its initialization and converges asymptotically to the true flow state value. Moreover, because of limited observations, noisy measurements produce noisy flow state estimated values. Both of these issues are not suitable in the framework of flow control since a poor and noisy estimated state used in a control law could trigger transition to turbulence in the controlled flow and therefore might cause the divergence of the control law [22], [23].

To deal simultaneously with the limited observations and the non-intrusive sensing issues, this paper proposes a vision-based control approach. As far as we know, such an approach has never been used for flow control issues. By using vision, dense flow velocity fields can be extracted from the flow image at video rate [24], [25], and used in an observer-free closed-loop scheme to control the flow. Closed-loop vision-based control is now a well established technique in the robotics and automatic control communities. Indeed this technique has shown impressive results in numerous complex contexts such as underwater, medical and aerial robotics [26]. Basically, this technique, also known as *visual servoing*, consists in using feedback information provided by a vision sensor to control a dynamic system [27].

In this paper, we apply the vision-based approach to the regulation of a plane Poiseuille flow around its steady state. Indeed this flow has become a standard problem to develop flow control theories [15], [16], [28], [29]: one of the main reason is that the analytical solution to the Navier-Stokes equations (NSE) of the steady state Poiseuille flow is well-known in fluid mechanics. However, as will be seen in Section II-C, the modeling of this flow for control issues relies on a conceptual model [15]. Unfortunately, it is almost impossible, or at least very difficult, to build an experimental

setup based on this model. Therefore, the scope of this paper is limited to a theoretical introduction of the vision-based control approach for fluid flows. Here we do not deal with practical implementation issues.

This paper is organized as follows: we first present the control-oriented modeling of plane Poiseuille flow. We continue by describing in Section III the two classical problems related to the control of plane Poiseuille flow. In addition we present the existing Poiseuille flow shear stress based control approaches which include the Linear Quadratic Gaussian (LQG) regulator. We then introduce in Section IV the fundamentals of the closed-loop control vision-based approach and we apply this approach to the control of Poiseuille flow in Section V. Since the shear stress based LQG regulator is the standard effective approach for flow control, in Section VI, we compare the shear stress based LQG approach with the proposed vision-based approach: we show that the shear stress based LQG approach is sensitive to measurements noise whereas the vision-based approach is very robust to noisy measurements. Finally, in Section VII the proposed theoretical results are validated in simulation using synthetic data sets of spatio-temporal variations of the perturbation velocities.

II. PLANE POISEUILLE FLOW MODELING FOR CONTROL DESIGN

In this section we first present the basics of plane Poiseuille flow, then we recall the boundary control principle for this flow and finally we present the reduced linearized model used to derive the currently existing control laws. Note that this modeling will also be used in part by the vision-based approach.

A. Basics

Poiseuille flow is a flow in an infinite length channel due to a pressure gradient. The non dimensionalized NSE of this flow are given by

$$\begin{cases} \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla)\mathbf{V} = -\nabla P + \frac{1}{R_e}\nabla^2 \mathbf{V} \\ \nabla \cdot \mathbf{V} = 0 \\ \mathbf{V}(x, y = \pm 1, z, t) = \mathbf{0} \end{cases} \quad (1)$$

where P is the pressure and $\mathbf{V}$ is the flow velocity. The x-axis is associated to the streamwise direction, the y-axis to the normal direction and the z-axis to the spanwise direction. $\mathbf{V}(x, y = \pm 1, z, t) = \mathbf{0}$ represents the no slip boundary condition and R_e is the Reynolds number.

Since Poiseuille flow is simple, the analytical solution $(V_{bx}, V_{by}, V_{bz}, P_b)$ of (1) in the steady state case, i.e. $\frac{d\mathbf{V}}{dt} = \frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla)\mathbf{V} = \mathbf{0}$, can be found:

$$(V_{bx}, V_{by}, V_{bz}, P_b) = (1 - y^2, 0, 0, -\frac{2}{R_e}x). \quad (2)$$

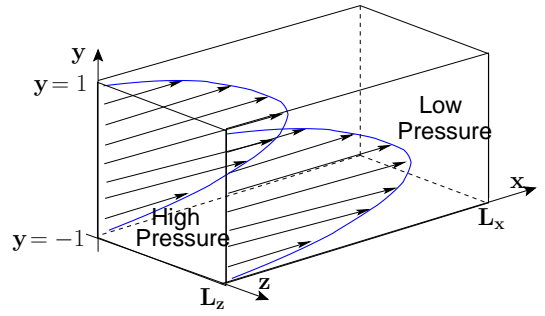The steady state velocities profile is illustrated on Fig. 1.

Fig. 1. Steady state velocities profile of a 3D plane Poiseuille flow: L_x and L_z are the streamwise and the spanwise lengths respectively.

B. Control principle

A perturbed plane Poiseuille flow can be controlled via boundaries. Boundary control consists in modifying boundaries conditions either on the lower boundary $y = -1$ [15] or on the upper boundary $y = 1$, or on both the upper $y = 1$ and lower $y = -1$ boundaries [28]. The exiting boundary control approach proposed for Poiseuille flow ensures mass conservation as shown in Fig. 2 where χ_u and χ_l are the boundary control functions of the upper and the lower channels, which can be interpreted as a geometric alteration of the boundaries. Note that in the absence of control, i.e. when $\chi_u = \chi_l = 0$, the red dashed curves (see Fig. 2) are aligned with the lower and upper boundary lines as expected.

From the existing boundary control approach of Poseuille flow, it is clear that the only appropriate actuator type which ensures mass conservation is a synthetic jet actuator since it is a zero net mass flux device [30]. Synthetic jet actuators are types of fluidic devices (mean to inject and to suck fluid). Note that fluidic devices are only one possible type of actuators for flow control since there are numerous other actuator control mechanisms such as plasma or other type which involves moving the domain boundary as described in [31].
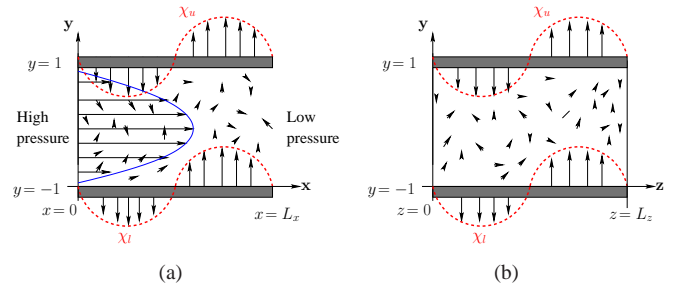
Fig. 2. Boundaries control of a 3D plane Poiseuille flow: (a) control viewed in the x-y plane, (b) control viewed in the y-z plane.

C. Reduced linearized model

For a practical implementation of flow control methods, the infinite dimension of a flow prompts the need for a reduced flow model. This section aims at deriving the reduced model of a controlled Poiseuille flow.

Concerning Poiseuille flow, most of the works focus on temporal instabilities caused by a perturbation velocity

$\mathbf{V_p}(x, y, z, t)$ defined as

$$\mathbf{V_p}(x, y, z, t) = \mathbf{V}(x, y, z, t) - \mathbf{V_b}(x, y, z, t) \qquad (3)$$

where $\mathbf{V_b}(x, y, z, t) = (V_{bx}, V_{by}, V_{bz})$ (see (2)). In order to keep permanent such instabilities in the infinite channel when the flow is not controlled, a periodic boundary finite length channel is assumed [15]. That is why the perturbation velocity $\mathbf{V_p}(x, y, z, t)$ can be expanded in a Fourier series

$$\mathbf{V_p}(x, y, z, t) = \sum_{n=-\infty}^{+\infty} V_p^n(y, t) e^{i(\alpha_n x + \beta_n z)}, \qquad (4)$$

where $(\alpha_n = n\alpha_0, \beta_n = n\beta_0)$, with $(\alpha_0 = \frac{2\pi}{L_x}, \beta_0 = \frac{2\pi}{L_z})$ the fundamental wavenumber pair and, L_x and L_z the streamwise and the spanwise period lengths respectively (see Fig. 1).

The modeling, required to derive existing control laws, consists first of all in linearizing the NSE around the steady state solution (2). Then the continuous linearized model of the NSE of the flow is reduced by approximation of the perturbation velocity $\mathbf{V_p}(x, y, z, t)$ at a specifically selected wavenumber pair (α_n, β_n) of the Fourier series (4); and by decomposition of the specifically selected Fourier series coefficient $V_p^n(y, t)$ through the evaluation of combinations of Chebychev polynomials Φ_m at Gauss-Lobatto collocation points y_k as follows

$$V_p^n(y_k, t) = \sum_{m=1}^{M} p_m^n(t) \Phi_m(y_k), \text{ with } 1 \leq k \leq M. \qquad (5)$$

This second step transforms a system of partial differential equations (PDE) obtained from (1) into a system of first order ordinary differential equations (ODE).

Finally, the null boundary conditions of the closed-loop control system is obtained by setting the upper and lower boundaries to the values of the control inputs χ_u and χ_l respectively. The details of the derivation of the reduced linearized model are given in Appendix A.

All computation done, the reduced linearized model is given by the following canonical expression

$$\begin{cases} \dot{\mathbf{p}}^{\mathbf{n}}(t) = \mathbf{A}^{\mathbf{n}}\mathbf{p}^{\mathbf{n}}(t) + \mathbf{B}^{\mathbf{n}}\mathbf{u}(t) \\ \mathbf{z}(t) = \mathbf{C}^{\top \mathbf{n}}\mathbf{p}^{\mathbf{n}}(t) \\ \mathbf{p}^{\mathbf{n}}(0) = \mathbf{p}^{\mathbf{n}}{}_0 \end{cases} \qquad (6)$$

where $\mathbf{p}^{\mathbf{n}}(t) = (p_m^n)$ is the state vector, $\mathbf{A}^{\mathbf{n}}$ is the state matrix, $\mathbf{u}(t) = (u_u(t), u_l(t))$ is the system control inputs on the upper and lower channel boundaries, like blowing or suction actions as proposed for instance in [15], $\mathbf{B}^{\mathbf{n}}$ is the input matrix, $\mathbf{C}^{\mathbf{n}}$ is the output matrix and $\mathbf{z}(t)$ is the vector of shear stress measurements on the upper and lower boundaries.

Using the reduced linearized model (6), we now review the two standard problems related to plane Poiseuille flow. We also present the existing solutions to these problems.

III. TWO CLASSICAL PROBLEMS AND THE EXISTING SOLUTIONS

In the framework of Poiseuille flow control, two main problems have been reported in the literature: the first problem concerns the case where the flow is unstable [32]; and the

second problem is related to the case where the kinetic energy density growth of the flow can instigate transition to turbulence [33]. We present these two problems below. We also present existing shear stress based control solutions to these problems.

A. First problem: unstable flow

For the Reynolds number $Re = 10\,000$, the wavenumber pair $(\alpha_n = 1, \beta_n = 0)$ is the only one (in the Fourier expansion (4)) which presents an unstable mode as proven by the solutions of the classical Orr-Sommerfeld equation [32]. This instability can be seen through the poles of the state matrix $\mathbf{A}^{\mathbf{n}}$, obtained by selecting the reduced linearized model of Poiseuille flow (6) at wavenumber pair $(\alpha_n = 1, \beta_n = 0)$. These poles are illustrated on Fig. 3, in particular the unstable poles $\lambda = 0.00373967 \pm i0.23752649$ are pointed out.
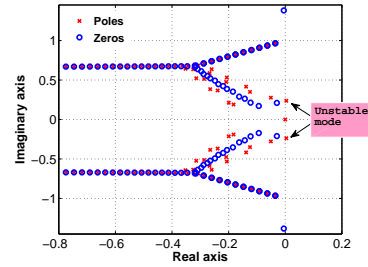
Fig. 3. Poles and zeros of the reduced linearized system for $Re = 10\,000$, $(\alpha_n = 1, \beta_n = 0)$; note that a complex conjugate pair of poles represents a mode.

In this case the flow is initially in the steady state but in an unstable equilibrium, i.e. a small disturbance velocity value $\mathbf{V_p}(x, y, z, t)$ destabilizes the uncontrolled fluid flow. Since $(\alpha_n = 1, \beta_n = 0)$, it is worth mentioning that the distribution of the flow velocities is the same in any z plane i.e.

$$\mathbf{V_p}(x, y, z, t) = V_p^n(y, t) e^{i(\alpha_n x)} , \forall z. \qquad (7)$$

It is therefore obvious to see that the reduced linearized model (6) of the unstable 3D plane Poiseuille flow can be obtained from a single 2D plane $z = z_p$ of the 3D flow. That is why in the sequel of this paper, instead of dealing with the unstable 3D flow, we choose, as in [15], [16], [29], [28], to address the problem of the unstable 2D plane Poiseuille flow destabilized by the perturbation velocity $\mathbf{V_p}(x, y, t)$.

B. Second problem: transient energy growth

The kinetic energy density of the flow perturbation $\mathbf{V_p}(x, y, z, t)$ is given by

$$\mathcal{E}_n(t) = \frac{1}{V_o} \int_{V_o} \frac{\|\mathbf{V_p}(x, y, z, t)\|^2}{2} \, dV_o, \qquad (8)$$

where V_o is the volume of a period of the domain under consideration. Plugging into (8) the above described Fourier (4) and Chebychev approximations (5) of the perturbation velocity $\mathbf{V_p}(x, y, z, t)$, it is possible to obtain a weighting matrix $\mathbf{Q}$ exactly as in [22] such that the kinetic energy density reduces to

$$\mathcal{E}_n(t) = \mathbf{p}^{\mathbf{n}\prime}(t) \mathbf{Q} \mathbf{p}^{\mathbf{n}}(t), \qquad (9)$$

where $\mathbf{x}'$ denotes the conjugate transpose of vector $\mathbf{x}$.

In the case where the Reynolds number is set to $R_e = 5000$ and the wavenumber pair is set to $(\alpha_n = 0, \beta_n = 2.044)$ the reduced linearized system is stable since all the poles of the state matrix $\mathbf{A^n}$ lay on the left part of the imaginary axis as shown on Fig. 4(a). However, in this case, it is possible to find the worst initial condition which causes the reduced linearized system (6) to present the maximum transient energy growth [33]. Indeed, a small perturbation
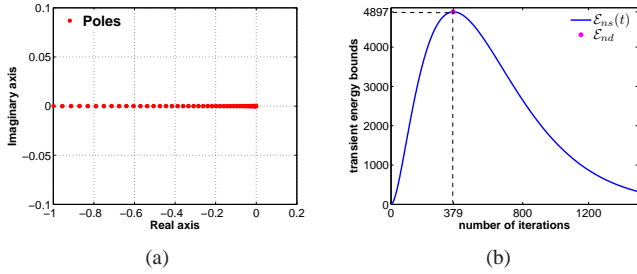

(a)   (b)

Fig. 4. 3D flow characteristics $R_e = 5000$, $(\alpha_n = 0, \beta_n = 2.044)$: (a) poles of the reduced linearized system, (b) synchronic and diachronic transient energy bounds.

velocity value $\mathbf{V_p}(x, y, z, t)$ in the reduced linearized system leads to a transient effect which is characterized by a growth in a short-time behaviour of the kinetic density energy, before a decay occurs. This transient effect, if not controlled, can cause transition to turbulence in the flow as explained in the following.

The synchronic transient energy bound at a given time t is defined as the maximum value of the kinetic energy density at time t:

$$\mathcal{E}_{ns}(t) = \max_{\mathcal{E}_n(0)=1} \mathcal{E}_n(t), \tag{10}$$

where $\mathcal{E}_n(t)$ is the kinetic energy density given in (9). The diachronic transient energy bound or the maximum transient energy growth is the maximum value of the synchronic transient energy bound over all time defined by

$$\mathcal{E}_{nd} = \max_{t \geq 0} \mathcal{E}_{ns}(t). \tag{11}$$

The synchronic and diachronic transient bounds are shown on Fig. 4(b). Let $\mathbf{p^n}_{\text{worst}}(0)$ be the initial condition corresponding the maximum transient energy growth shown with the magenta dot on Fig. 4(b). As already mentioned above, if no control is applied when $\mathbf{p^n}_{\text{worst}}(0)$ is the system initial condition, then the resulting highest transient energy growth could instigate transition to turbulence in the flow.

C. Existing solutions: shear stress based control

Using the classical output feedback control $u_j(t) = -k_j z_j(t)$ in (6), where u_j is the single control input, k_j is a scalar gain and $z_j(t)$ is the shear stress measurement at a single point, the unstable 2D Poiseuille flow can be stabilized as shown in [15]. However this simple proportional controller generally fails to suppress unobservable high transient modes which could trigger transition to turbulence [15]. By supposing the availability

of the value of the state vector $\mathbf{p^n}(t)$, the observability is assumed and a state feedback Linear Quadratic Regulator (LQR), can easily stabilize the unstable perturbed Poiseuille flow, and can reduce the growth of the transient energy limiting thus the risk of an excursion in the turbulent state [16], [28]. Considering an infinite time horizon, the LQR signal

$$\mathbf{u}(t) = -\mathbf{k}^\top \mathbf{p^n}(t) \tag{12}$$

minimizes the cost function

$$\int_0^\infty \left(\mathbf{p^{n'}}(t)\mathbf{Q}\mathbf{p^n}(t) + \mathbf{u}'(t)\mathbf{R}\mathbf{u}(t) \right) \mathrm{d}t \tag{13}$$

where $\mathbf{k}$ is the optimal gain, $\mathbf{Q}$ and $\mathbf{R}$ are positive-semidefinite weighting matrices. Matrix $\mathbf{R}$ is used to limit energy consumption of actuators. In order to maintain wall symmetry, matrix $\mathbf{R}$ is set as a scaled identity matrix [22], i.e. $\mathbf{R} = r^2\mathbf{I}$, where r is a real positive parameter. Note that by choosing matrix $\mathbf{Q}$ as the same matrix involved in expression of the kinetic energy density (see (9)), this control law can reduce the growth of transient energy density limiting thus the risk of an excursion in the turbulent state [16], [28].

However, in practice the LQR approach can not be used since it requires the unknown value of the true state value $\mathbf{p^n}(t)$. A Linear Quadratic Gaussian (LQG) controller is thus required. This approach is based on an estimated value $\widehat{\mathbf{p^n}}(t)$ of the state vector. The value of $\widehat{\mathbf{p^n}}(t)$ is obtained from the shear stress measurements $\mathbf{z}(t)$ using an observer built from a Linear Quadratic Estimation (LQE) scheme (see Section VI-A). Instead of using (12), the control signal for the output feedback LQG regulator is given by

$$\mathbf{u}(t) = -\mathbf{k}^\top \widehat{\mathbf{p^n}}(\mathbf{z}(t)), \tag{14}$$

where vector $\mathbf{k}$ is still the LQR optimal gain and $\widehat{\mathbf{x}}$ represents an estimate of the value of $\mathbf{x}$. This last control law (14) will be refer to as shear stress based LQG (SSB-LQG) control.

However as already mentioned in Section I, any observer suffers from the initialization issue as shown in Section VI-A. In addition in the case where the limited shear stress measurements $\mathbf{z}(t)$ are noisy, the estimate $\widehat{\mathbf{p^n}}(t)$ is also noisy as shown in Section VI-A. Both of these problems are solved by the proposed vision-based control approach, which is introduced in the next section.

IV. FUNDAMENTALS OF VISUAL SERVOING

As already mentioned in Section I, visual servoing is a well-known approach in the robotics and automatic control communities for non linear control of complex systems. This approach consists in using feedback information from a vision sensor to control the state of a dynamic system [27]. To achieve a vision-based control task, a set of visual features $\mathbf{s}(t)$ is selected from the image of the scene. Indeed, only a part of the image (provided by the sensor) is used to define a diffeomorphic map between the observed scene and a judiciously selected set of features in the image. A control law is then designed so that the visual features $\mathbf{s}(t)$ reaches a desired value $\mathbf{s}^*$ corresponding to a desired state of the

system. The control principle is thus to regulate the error vector $\mathbf{e}(t) = \mathbf{s}(t) - \mathbf{s}^*$ to zero.

To design the control law, the dynamic of the error vector must be known. This dynamic is given by

$$\dot{\mathbf{e}}(t) = \frac{\partial \mathbf{e}(t)}{\partial t} + \mathbf{L_e}(t)\mathbf{u}(t), \qquad (15)$$

where $\mathbf{u}(t)$ is the system control inputs, $\mathbf{L_e}(t)$ is the jacobian matrix that encodes the time variation of the visual features with respect to the variation of the control signal acting on the system [34], and $\partial \mathbf{e}(t)/\partial t$ expresses the variation of the error vector due to the free motion of the visual features.

A key point in vision-based control is that this control technique belongs to the class of sensor-based control of dynamic systems: the control law is computed in the sensor frame [35]. Consequently, this approach corresponds clearly to an observer-free feedback control.

In the following we apply the visual servoing approach to the regulation of plane Poiseuille flow.

V. VISUAL SERVOING FOR FLOW CONTROL

In the particular case of flow control, a control law is designed from visual features obtained from the vision system sensing the flow. Of course a great advantage of such a sensor is that it is non-intrusive. This sensor is also an extremely rich and dense source of information on the flow. Indeed a large spectrum of visual features $\mathbf{s}(t)$ could be selected from the image, such as coordinates of singular points in the flow or the vorticity map. Nevertheless, to directly compare our vision-based approach to the SSB-LQG approach we choose the same data, i.e. $\mathbf{s}(t) = \widehat{\mathbf{p^n}}(t)$. However, as shown in the next section, the main difference between both approaches is the way to estimate $\widehat{\mathbf{p^n}}(t)$. We first show in our approach how $\widehat{\mathbf{p^n}}(t)$ is estimated from visual measurements, and then we present the control law.

A. State estimation from visual measurements

Here we consider both the control of the 2D and the 3D plane Poiseuille flows.

1) 2D plane Poiseuille flow: A laser sheet is used to enlighten the particles for which the velocities are computed (see Fig. 5). Consequently, from this visualization process, it is possible to compute dense flow velocity maps from optical flow techniques. Optical flow can be defined as the apparent velocity vector field representing the motion of photometric pattern (pixels brightness) in successive image sequences [36]. We first present the perspective projection of a flow particle, then we show how to estimate a flow particle velocity from its image velocity, and finally we present the computation of the state vector from the velocity of a flow particle.

a) Perspective projection of a flow particle: Let $^o\mathbf{M} = \left(^oM_x(t), \,^oM_y(t), \,^oM_z(t)\right)$ be the space-time coordinates of the flow particle $\mathcal{M}$ expressed in the flow frame $\mathcal{F}_o$. The perspective projection of $\mathcal{M}$ is obtained in three steps.

- The first step consists to express $\mathcal{M}$ in the camera frame. Let $^c\mathbf{M} = \left(^cM_x(t), \,^cM_y(t), \,^cM_z(t)\right)$ be the space-time

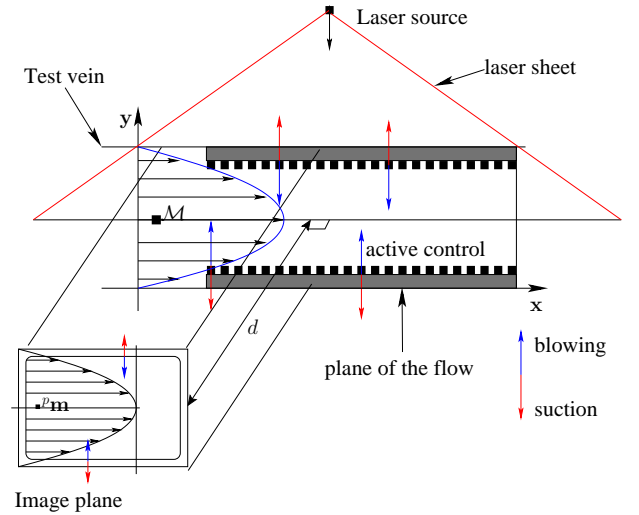Fig. 5. Fronto-parallel visualization of a 2D flow using a laser sheet which role is to enlighten the particles seeded in the fluid: $^p\mathbf{m}$ is the perspective image of the fluid particle $\mathcal{M}$ in the sensor space.
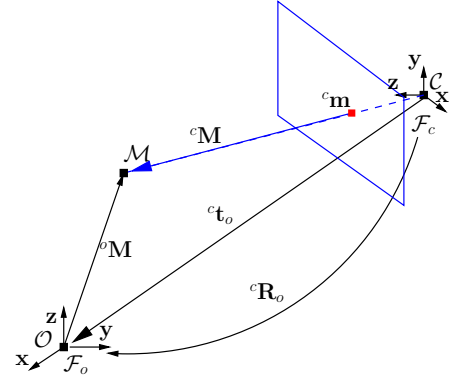

Fig. 6. Perspective projection of a flow particle.

coordinates of $\mathcal{M}$ expressed in the camera frame $\mathcal{F}_c$. The relationship between $^c\mathbf{M}(t)$ and $^o\mathbf{M}(t)$ is given by

$$^c\mathbf{M}(t) = \,^c\mathbf{R}_o\,^o\mathbf{M}(t) + \,^c\mathbf{t}_o \qquad (16)$$

where $(^c\mathbf{R}_o, \,^c\mathbf{t}_o)$ is the rigid constant kinematic link between the camera and the flow frames (see Fig. 6). This rigid link is also known as extrinsic camera parameters.

- In the second step, the perspective projection $^c\mathbf{m}(t) = (^cm_x(t), \,^cm_y(t))$ of point $^c\mathbf{M}(t)$ obtained from the well-known pinhole camera model is given by:

$$^cm_x(t) = \frac{^cM_x(t)}{^cM_z(t)}, \quad ^cm_y(t) = \frac{^cM_y(t)}{^cM_z(t)}. \qquad (17)$$

More details on perspective projection models can be found in [37].

- Finally in the last step, the perspective coordinates vector $^c\mathbf{m}(t)$ is expressed in the sensor space (i.e. in pixel unit) as $^p\mathbf{m}(t) = (^pm_x(t), \,^pm_y(t))$, by the relation

$$^pm_x(t) = \frac{f}{l_u}\,^cm_x(t) + u_0, \quad ^pm_y(t) = \frac{f}{l_v}\,^cm_y(t) + v_0 \qquad (18)$$

where l_u (respectively l_v) is the pixel size (in meter) in the u (respectively v) direction, f is the fo-

cal length and (u_0, v_0) is the vector coordinates of the principal point of the camera. Note that vector $(f_u = f/l_u, f_v = f/l_v, u_0, v_0)$ represents the intrinsic camera parameters. Both the intrinsic and the extrinsic camera parameters can be determined using the calibration method described in [38].

b) Estimation of the flow velocity particle from its image velocity: Now we show in three steps how to compute the velocity of a flow particle $\mathcal{M}$ from its perspective image $^p\mathbf{m}$. The first step consists in expressing the relationship between the flow particle and its perspective image velocities. From (18), it is easy to show that the relationship between the image velocities $^p\dot{\mathbf{m}}$ and the flow velocities $^o\dot{\mathbf{M}}(t) = \begin{bmatrix} ^o\dot{M}_x(t) \\ ^o\dot{M}_y(t) \\ ^o\dot{M}_z(t) \end{bmatrix}$ is given by

$$^p\dot{\mathbf{m}}(t) = \frac{\partial ^p\mathbf{m}}{\partial ^c\mathbf{M}} {}^c\mathbf{R}_o {}^o\dot{\mathbf{M}} \qquad (19)$$

where the jacobian matrix $\frac{\partial ^p\mathbf{m}}{\partial ^c\mathbf{M}}$ is given by

$$\frac{\partial ^p\mathbf{m}}{\partial ^c\mathbf{M}} = \begin{bmatrix} \frac{f_u}{M_z(t)} & 0 & -\frac{f_u M_x(t)}{M_z^2(t)} \\ 0 & \frac{f_v}{M_z(t)} & -\frac{f_v M_y(t)}{M_z^2(t)} \end{bmatrix}. \qquad (20)$$

In the 2D case, since $^oM_z(t)$ is constant, we have $^o\dot{M}_z(t) = 0$ and (19) can be rewritten as

$$\begin{bmatrix} ^p\dot{m}_x(t) \\ ^p\dot{m}_y(t) \end{bmatrix} = \frac{\partial ^p\mathbf{m}}{\partial ^c\mathbf{M}} {}^c\mathbf{R}_o|_{12} \begin{bmatrix} ^o\dot{M}_x(t) \\ ^o\dot{M}_y(t) \end{bmatrix} \qquad (21)$$

where $\mathbf{R}_o|_{12}$ is the restriction of the orientation matrix $^c\mathbf{R}_o$ to its two first columns.

The second step consists to estimate the image velocity $^p\dot{\mathbf{m}}$. Vector $^p\dot{\mathbf{m}}$ can be determined by solving a matching problem between two consecutive images. Two approaches exist to solve this matching problem: local and global approaches.

i) Local approaches: region-based matching
The goal is to estimate the displacement of the point $^p\mathbf{m}(t)$ between two images acquired at a very short time interval by comparing windows (local regions). This comparison can be done by:

- maximizing a similarity measurement, such as the cross-correlation used in the PIV community;
- minimizing a dissimilarity measurement, such as the sum-of-squared differences used in the computer vision community;
- using a variational approach. In this case, the optical flow is computed. The optical flow equation expresses the brightness conservation at pixel $^p\mathbf{m}(t)$ as follows

$$0 = \frac{dI_m(t)}{dt} = \frac{\partial I_m(t)}{\partial t} + \nabla I_m^\top {}^p\dot{\mathbf{m}}, \qquad (22)$$

where $I_m(t)$ is the brightness of pixel $^p\mathbf{m}(t)$ and ∇I_m is the brightness spatial gradient. From (22), assuming that $^p\dot{\mathbf{m}}(t)$ is constant within a neighbourhood $\mathcal{V}(^p\mathbf{m})$ of $^p\mathbf{m}(t)$, $^p\dot{\mathbf{m}}(t)$ can be computed using the well-known Lucas-Kanade method based on a least-squares solution [39].

ii) Global approaches: prior regularity model on motion
On the entire image, the goal is to solve a minimization problem composed of two terms

$$f(I_m, \mathbf{v}) = f_d(I_m, \mathbf{v}) + \alpha f_r(\mathbf{v}) \qquad (23)$$

where $\mathbf{v}$ is the velocity field to estimate, and where

- $f_d(I_m, \mathbf{v})$ is the data term or the observation model that enforces the conservation assumption for instance
  - the brightness conservation

$$f_d(I_m, \mathbf{v}) = \frac{1}{2} \int_\Omega \frac{\partial I_m}{\partial t} + \nabla I_m^\top \mathbf{v} \, ds \qquad (24)$$

  where Ω is the domain of the flow in the image; or
  - a physics-based optical flow equation for laser sheet illuminated particles in a volume and planar control surfaces [40]

$$f_d(I_m, \mathbf{v}) = \frac{1}{2} \int_\Omega \left(\frac{\partial I_m}{\partial t} + \nabla I_m^\top \mathbf{v} + I_m \mathrm{div}\mathbf{v} \right)^2 ds; \qquad (25)$$

- $f_r(\mathbf{v})$ is the regularization term which enforces a spatial smoothness of the minimizing velocity $\mathbf{v}$ to a degree prescribed by the regularization parameter α. For a first order regularization $f_r(\mathbf{v})$ can be chosen as [41]:

$$f_r(\mathbf{v}) = \frac{1}{2} \int_\Omega \|\nabla \mathbf{v}_x\|^2 + \|\nabla \mathbf{v}_y\|^2 \, ds; \qquad (26)$$

and for the second order regularization $f_r(\mathbf{v})$ can be chosen as [42], [43]:

$$f_r(\mathbf{v}) = \frac{1}{2} \int_\Omega \left(\|\nabla \mathrm{div}\mathbf{v}\|^2 + \|\nabla \mathrm{curl}\mathbf{v}\|^2 \right) ds; \qquad (27)$$

This second matching approach provides a dense vector field (one per pixel) with spatial coherence.

To sum up, the flow velocity in the image $^p\dot{\mathbf{m}}$ can be estimated using either a region-based matching approach that provides velocity field over local image regions at video rate [24], [25], [39]; or a global matching approach. This last matching approach, although not running at video rate because of current hardware limitations, provides dense velocity map, i.e. one velocity per pixel, with spatial coherence. A comprehensive review of some NSE consistent optical flow methods is available in [36].

Finally, assuming a perfectly calibrated camera, it becomes straightforward to express the flow perturbation velocities $\widehat{\mathbf{V}_\mathbf{p}}(x, y, t)$ from optical flow measurements by inverting (21) and by using (3).

c) Estimation of the state vector: This last step shows how to compute $\widehat{\mathbf{p^n}}(t)$ from the estimation $\widehat{\mathbf{V}_\mathbf{p}}(x, y, t)$. This is done by projecting the perturbation velocities $\widehat{\mathbf{V}_\mathbf{p}}(x, y, t)$ onto Fourier and Chebychev bases as detailed in Appendix B-A.

It is worth mentioning that an important contribution of this method relies on the fact that the initial value $\widehat{\mathbf{p^n}}(t = 0)$ is therefore no longer of concerned in our approach as shown in Section VI-B. In addition, because of dense flow velocity maps, the vision-based approach provides less noisy estimations of the state vector as shown also in Section VI-B.

We now focus on the 3D case by showing how to obtain the state vector $\widehat{\mathbf{p^n}}(t)$ from visual measurements.

2) 3D plane Poiseuille flow: We first show the estimation of $\widehat{\mathbf{V_p}}(x, y, z, t)$ from visual measurements and then we compute the state vector $\mathbf{p^n}(t)$ from $\widehat{\mathbf{V_p}}(x, y, z, t)$

a) Estimation of the flow velocity particle from its image velocity: The complete 3D flow perturbation velocity $\widehat{\mathbf{V_p}}(x, y, z, t)$ can be reconstructed in a volume using a stereoscopic pair of images of the flow $\mathbf{I}_1$ and $\mathbf{I}_2$. Indeed, let $^p\mathbf{m}_1$ and $^p\mathbf{m}_2$ be the projection of the 3D point $^o\mathbf{M}$ points in images $\mathbf{I}_1$ and $\mathbf{I}_2$ respectively. Using (19) in both images $\mathbf{I}_1$ and $\mathbf{I}_2$, stereoscopic particle image velocimetry (stereo-PIV) technique can be used to compute $\mathbf{V}(x, y, z, t)$ and thus $\widehat{\mathbf{V_p}}(x, y, z, t)$ in a planar domain [44]. Based on stero-PIV technique, it is possible to devise an automatic high speed optical scanner which provides many different planes of the flow, leading thus to the estimation of a 3D volume of the observed flow [45]. It is also possible to observe the flow with three cameras providing thus two different planes of view from which the complete velocity gradient tensor can be computed: this technique is known as dual-plane PIV [46].

b) Estimation of the state vector: As for the 2D case, $\mathbf{p^n}(t)$ can be obtained by projecting the estimation $\widehat{\mathbf{V_p}}(x, y, z, t)$ onto the Fourier and Chebychev bases as detailed in Appendix B-B. Of course, here again, $\widehat{\mathbf{p^n}}(t)$ does not depend on $\widehat{\mathbf{p^n}}(t = 0)$ as it is the case (demonstrated in Section VI-A) with the SSB-LQG approach. Since $\widehat{\mathbf{p^n}}(t)$ is available in both the 2D and 3D cases, the control law can now be derived.

B. Closed-loop vision-based control of flows

As mentioned at the very beginning of this Section V, we choose the same data $\mathbf{s}(t) = \widehat{\mathbf{p^n}}(t)$ as in the SSB-LQG approach so that our vision-based approach can be directly compare to the SSB-LQG approach. In that case, it becomes easy to express the dynamic of the error of the visual features $\mathbf{e}(t)$ (see (15)) around the steady state solution (the equilibrium point). In the ideal case where there is no state disturbances and there is no image noise, from the ideal state dynamic equation given in (6), it is straightforward that the ideal image error dynamic given by (15) is such that

$$\frac{\partial \mathbf{e}(t)}{\partial t} = \mathbf{A^n} \mathbf{e}(t), \quad \mathbf{L_e}(t) = \mathbf{B^n} \text{ with } \mathbf{e}(t) = \mathbf{p^n}(t). \quad (28)$$

More precisely (28) can be physically interpreted as follows: the term $\frac{\partial \mathbf{e}(t)}{\partial t}$ describes the instationary aspect of the flow image velocity map due to the motion of the uncontrolled flow, and the term $\mathbf{L_e}(t)$ encodes the spatial inhomogeneity aspect of the flow image velocity map due to unsteady actuation (for instance blowing and suction actions).

Consequently, around the desired state, (15) can be rewritten as

$$\dot{\mathbf{p}}^\mathbf{n}(t) = \mathbf{A^n}\mathbf{p^n}(t) + \mathbf{B^n}\mathbf{u}(t) \quad (29)$$

and the simple state-feedback control law

$$\mathbf{u}(t) = -\mathbf{k}^\top \widehat{\mathbf{p^n}}(t), \quad (30)$$

can be used. We will refer to this vision-based control law as the vision-based LQG (VB-LQG) control law since $\widehat{\mathbf{p^n}}(t)$

is obtained from visual measurements instead of shear stress measurements as used in (14).

In the next section, we compare the SSB-LQG approach with the proposed VB-LQG method.

VI. COMPARISON BETWEEN THE SSB-LQG AND THE VB-LQG APPROACHES

The major difference between the SSB-LQG and the VB-LQG approaches is the estimation of the state vector $\widehat{\mathbf{p^n}}(t)$: the SSB-LQG regulator uses an observer built from the LQE approach whereas the VB-LQG approach relies on an observer-free estimation method. We now highlight the influence of these estimation methods on the closed-loop system.

A. Behaviour of the system closed by the SSB-LQG control law

We first present the LQE framework and then we present the system closed by the SSB-LQG control. In the LQE framework, on which the SSB-LQG approach relies, it is generally assumed that the reduced linearized system (6) has process disturbances $\varepsilon_\mathbf{p}$ and measurements noise ε_z. In addition $\varepsilon_\mathbf{p}$ and ε_z are assumed to be uncorrelated Gaussian white noise with covariance matrices $\Xi_\mathbf{p}$ and Ξ_z respectively. The reduced linearized system (6) is thus rewritten as

$$\begin{cases} \dot{\mathbf{p}}^\mathbf{n}(t) = \mathbf{A^n}\mathbf{p^n}(t) + \mathbf{B^n}\mathbf{u}(t) + \varepsilon_\mathbf{p}(t) \\ \mathbf{z}(t) = \mathbf{C}^{\top \mathbf{n}}\mathbf{p^n}(t) + \varepsilon_z(t) \\ \mathbb{E}\{\varepsilon_\mathbf{p}\varepsilon'_\mathbf{p}\} = \Xi_\mathbf{p}, \ \mathbb{E}\{\varepsilon_z\varepsilon'_z\} = \Xi_z \\ \mathbb{E}\{\varepsilon_z\varepsilon'_\mathbf{p}\} = \mathbf{0}, \ \mathbb{E}\{\varepsilon_\mathbf{p}\varepsilon'_z\} = \mathbf{0} \\ \mathbf{p^n}(0) = \mathbf{p^n}_0, \end{cases} \quad (31)$$

where $\mathbb{E}\{\cdot\} = \lim_{T \to \infty} \frac{1}{T} \int_0^T \{\cdot\} \, \mathrm{d}t$ is the expectation operator.

Let $\boldsymbol{\delta}^\mathbf{n}(t) = \widehat{\mathbf{p^n}}(t) - \mathbf{p^n}(t)$ be the estimation error. From the plant model given in (31), a consistent estimation scheme is given by

$$\begin{cases} \dot{\widehat{\mathbf{p}}}^\mathbf{n}(t) = \mathbf{A^n}\widehat{\mathbf{p^n}}(t) + \mathbf{B^n}\mathbf{u}(t) + \mathbf{L}\varphi(t) \\ \varphi(t) = \mathbf{z}(t) - \mathbf{C}^{\mathbf{n}\top}\widehat{\mathbf{p^n}}(t) \\ \widehat{\mathbf{p^n}}(0) = \textbf{unknown}, \end{cases} \quad (32)$$

where $\widehat{\mathbf{p^n}}(t)$ is the estimated state vector for an infinite time horizon, and $\mathbf{L}$ the optimal gain that minimizes $\mathbb{E}\{[\boldsymbol{\delta}^\mathbf{n}(t)]'[\boldsymbol{\delta}^\mathbf{n}(t)]\}$ given by $\mathbf{L} = \mathbf{x}\mathbf{B}^{\mathbf{n}\top}\Xi_\mathbf{z}^\top$ where $\mathbf{x}$ (a positive-semidefinite matrix) is the solution of the algebraic Riccati equation $\mathbf{x}\mathbf{A}^{\mathbf{n}\top} + \mathbf{A^n}\mathbf{x} - \mathbf{x}\mathbf{C}^{\mathbf{n}\top}\Xi_z^{-1}\mathbf{C^n}\mathbf{x} + \Xi_\mathbf{p} = 0$. The term $\varphi(t)$ is the innovation term representing the difference between the current measurement and its prediction. The higher the value of $\mathbf{L}$ the more reactive the system.

Using (31) and (32) we obtain the estimation error dynamic

$$\begin{cases} \dot{\boldsymbol{\delta}}^\mathbf{n}(t) = \mathbf{A}_\mathbf{L}^\mathbf{n} \boldsymbol{\delta}^\mathbf{n}(t) + \mathbf{L}\varepsilon_z(t) - \varepsilon_\mathbf{p}(t) \\ \boldsymbol{\delta}^\mathbf{n}(0) = \textbf{unknown} \end{cases} \quad (33)$$

where $\mathbf{A}_\mathbf{L}^\mathbf{n} = \mathbf{A^n} - \mathbf{L}\mathbf{C}^{\mathbf{n}\top}$. The solution of (33) is

$$\boldsymbol{\delta}^\mathbf{n}(t) = \mathrm{e}^{\mathbf{A}_\mathbf{L}^\mathbf{n} t}\boldsymbol{\delta}^\mathbf{n}(0) + \int_0^t \mathrm{e}^{\mathbf{A}_\mathbf{L}^\mathbf{n}(t-\tau)}\big(\mathbf{L}\varepsilon_z(\tau) - \varepsilon_\mathbf{p}(\tau)\big)d\tau. \quad (34)$$

As shown in (34), the initial error value highly influences the convergence time after which $\widehat{\mathbf{p^n}}(t)$ equals the true value

of the state vector $\mathbf{p^n}(t)$: this is the well known asymptotic convergence property of the LQE. Modeling the initial condition $\widehat{\mathbf{p^n}}(t=0)$ with known physical statistics about the studied flow and using a time-varying estimator gain $\mathbf{L}(t)$ can reduce the time to which $\widehat{\mathbf{p^n}}(t)$ equals the true value of the state vector $\mathbf{p^n}(t)$ as proposed in [21]. But this solution is still not satisfactory since it needs additional a-priori parameters in the model of the initial condition. In addition, (34) shows that the estimation scheme also suffers from modeling error $\varepsilon_{\mathbf{p}}$ and measurements noise ε_z. Expression (34) also clearly shows that a compromise on the value of $\mathbf{L}$ has to be found: indeed the value of $\mathbf{L}$ must be low to limit the influence of the measurement noise while we have pointed out that the value of $\mathbf{L}$ must be high to take into account the innovation term $\varphi(t)$.

Now we present the system closed by the SSB-LQG control. By combining the estimation error dynamic given in (33) and the control law (14) into the first equation of (31)), the behaviour of the closed-loop system is given by

$$\begin{cases} \begin{bmatrix} \dot{\mathbf{p}}^{\mathbf{n}}(t) \\ \dot{\boldsymbol{\delta}}^{\mathbf{n}}(t) \end{bmatrix} = \mathbf{M}_{lqg} \begin{bmatrix} \mathbf{p^n}(t) \\ \boldsymbol{\delta^n}(t) \end{bmatrix} + \mathbf{M}_n \begin{bmatrix} \varepsilon_{\mathbf{p}}(t) \\ \varepsilon_z(t) \end{bmatrix} \\ \boldsymbol{\delta^n}(0) = \mathbf{unknown} \end{cases} \quad (35)$$

where $\mathbf{M}_{lqg} = \begin{bmatrix} \mathbf{A_k^n} & -\mathbf{B^n k}^\top \\ \mathbf{0} & \mathbf{A_L^n} \end{bmatrix}$, $\mathbf{M}_n = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{I} & \mathbf{L} \end{bmatrix}$ with $\mathbf{A_k^n} = \mathbf{A^n} - \mathbf{B^n k}^\top$ and $\mathbf{I}$ an identity matrix. Equation (35) clearly shows that the true state dynamic $\dot{\mathbf{p}}^{\mathbf{n}}(t)$ (around the desired state) depends on the estimation error. Therefore, since this error highly depends on the initial unknown estimation error, a poor initialization of the observer could drive the system to a turbulent state as shown in [22]. In addition the noise in the measurements propagates in the control law (see (14)), this is not suitable at all for the lifetime of the actuators.

B. Behavior of the system closed by the VB-LQG control law

We first present the estimation of the state vector $\widehat{\mathbf{p^n}}(t)$ in the case of noisy optical flow measurements only in the 2D case[1], and then we present the system closed by the VB-LQG control. In the case of the 2D plane Poiseuille flow, it is possible to express the estimation error $\boldsymbol{\delta^n}(t)$ when $\widehat{\mathbf{p^n}}(t)$ is estimated from optical flow measurements. Indeed, let $\varepsilon(x, y, t)$ be a 2D independent identically distributed white Gaussian process. Let $\mathbf{V_p}(x, y, t)$ be the $M \times N$ locations size image of the perturbation velocity map obtained from noise-free optical flow measurements. From (21), in the practical case where optical flow measurements $^p\dot{\mathbf{m}}$ are corrupted by a Gaussian noise process, the computed flow perturbation velocities $\mathbf{V_p}$ are also affected by Gaussian noise. Without lost of generality the noisy perturbation velocity map can

be written as $\widehat{\mathbf{V_p}}(x, y, t) = \mathbf{V_p}(x, y, t) + \varepsilon(x, y, t)$. This noisy perturbation velocity map $\widehat{\mathbf{V_p}}(x, y, t)$ is used to compute the estimation of the state vector $\mathbf{p^n}(t)$ as detailed in Appendix B-A. We obtain

$$\widehat{\mathbf{p^n}}(t) = \mathbf{p^n}(t) + \frac{1}{N}\mathbf{e^n}(t), \quad (36)$$

from which we immediately deduce the estimation error

$$\boldsymbol{\delta^n}(t) = \frac{1}{N}\mathbf{e^n}(t) \quad (37)$$

where vector $\mathbf{e^n}(t)$ the projection of the measurements noise matrix $\varepsilon(t)$ onto Fourier and Chebychev bases. It is clear from (36), that the larger the value of N, typically $N > 1024$ for real images, the smaller the state vector estimation error. Although for a PIV image (based on a correlation technique) the number of velocity measurements is less than the number of pixels in the image, a PIV image still provides sufficient measurements for N to be large enough to reduce the noise. Therefore, in the vision-based case, contrary to (34), the estimation error (37) does not depend anymore on the initial estimation error: this is a strong advantage of our approach.

We now present the system closed by the VB-LQG control. As for the SSB-LQG approach, the behaviour of the closed-loop system can be also obtained. In this case, the reduced linearized system (6) controlled by the vision-based approach (30) is given by

$$\begin{cases} \dot{\mathbf{p}}^{\mathbf{n}}(t) = \mathbf{A^n p^n}(t) + \mathbf{B^n u}(t) \\ \mathbf{u}(t) = -\mathbf{k}^\top \widehat{\mathbf{p^n}}(t) \\ \boldsymbol{\delta^n}(t) = \frac{1}{N}\mathbf{e^n}(t), \ \mathbf{p^n}(0) = \mathbf{p^n}_0, \end{cases} \quad (38)$$

leading to the following behaviour:

$$\dot{\mathbf{p}}^{\mathbf{n}}(t) = (\mathbf{A^n} - \mathbf{B^n k}^\top)\mathbf{p^n}(t) - \frac{1}{N}\mathbf{B^n k}^\top \mathbf{e^n}(t), \ \mathbf{p^n}(0) = \mathbf{p^n}_0. \quad (39)$$

The initial value $\widehat{\mathbf{p^n}}(t=0)$ is therefore no longer of concerned in our approach. In addition for a large number of velocity measurements N, the reduced linearized system dynamic equation (39) is less affected by measurements noise since $\frac{1}{N}\mathbf{B^n k}^\top \mathbf{e^n}(t)$ tends to $\mathbf{0}$. This is another great improvement over the SSB-LQG control scheme that is always noise dependent when noisy shear stress values are used in the LQE approach as shown in (35).

VII. RESULTS

In this section we validate the proposed theoretical results. The validations of our approach do not use real optical flow measurements for the 2D flow or PIV measurements for the 3D flow. We rather use synthetic data sets of spatio-temporal variations of the perturbation velocities obtained from the Poiseuille flow reduced linearized model presented in (6). Using Matlab codes provided in [22], matrices $\mathbf{A^n}$, $\mathbf{B^n}$, $\mathbf{C^n}$ given in (6) are computed. We first present the result for the stabilization of the 2D Poiseuille flow (see Section III-A) and then we present the result of the reduction of the transient energy growth in the 3D flow (see Section III-B). Finally we discuss about the potential advantages of the vision-based approach for flow control.

---

[1]Theoretically, because of the non-linear process that transforms the 2D image velocities to 3D flow particles velocities, the potential drawback with a stereo-PIV technique is that a small image processing error could lead to large error in the estimation of the velocities $\mathbf{V_p}$; In addition a stereo-PIV technique rely on an accurate calibration of two cameras. Modeling the effects of image processing and calibration errors is very complex. That is why we leave for future works the robustness analysis to noise of the vision-based state estimation for 3D flows. Nevertheless in practice the cameras are usually accurately calibrated and the image processing algorithms are robust to noise.

A. Stabilization of the unstable flow

First of all, the behaviour of the VB-LQG approach is shown in the ideal case, then we compare the estimation of the state vector provided by shear stress measurements or by optical flow measurements. Finally, the behaviour of the closed-loop systems are presented in both cases. The following classical characteristics have been used as in [15], [16] and [28]: the Reynolds number is $R_e = 10\ 000$, the length of the channel is $L_x = 4\pi$, and the reduced model Fourier wavenumber pair is $(\alpha_n = 1, \beta_n = 0)$. The parameter r of the weighting matrix $\mathbf{R}$ is $r = 200$, penalizing thus high control values.

1) VB-LQG control in the ideal case: We first present results concerning the VB-LQG control approach (30) in the ideal case where there is no measurements noise. Fig. 7 shows the different steps in the control of the perturbed flow with $N = 252$. Fig. 7(a) pictures the desired image of the flow corresponding to the steady state velocities profile; Fig. 7(b) shows the image of the flow just before the application of the vision-based control law where we can see that the flow has become turbulent. Figs. 7(c) and 7(d) show different steps of the controlled flow at arbitrary selected iteration numbers $k = 1047$ and $k = 1500$ respectively: the control at each selected instant is represented by green vertical arrows on the upper and the lower channel boundaries. The control law converges since it tends towards 0 as shown on Fig. 7(e). Moreover, Fig. 7(f) depicts the kinetic energy density of the flow perturbation where we can see an increase due to the perturbation growth in the case where the flow is not controlled; and then a decrease also towards 0 once the control law is applied. At this step, we can see that the final velocities profile given in Fig. 7(d) is very similar to the desired velocities profile in Fig. 7(a). Therefore, the VB-LQG approach performs as expected.

2) Comparison of the estimation methods: In this section we show that the vision-based state estimation provides better results than the LQE state estimation. We consider a perturbed flow which is not controlled. Results are given in Fig. 8 in terms of the square norm of the state vector instead of the more relevant $2M$ (normally greater than 40 for a more accurate reduced model) components of the state vector for the sake of clarity and readability. Fig. 8(a) presents the ideal case where there is no measurements noise and no initialization error. From this figure we can see that both estimations perfectly correspond to the ground truth value of the state vector. Fig. 8(b) highlights the poor initialization issue and the asymptotic convergence issue in the LQE; these issues are not of concerned in the vision-based approach which provides the ground truth value of the state vector. This result confirms that the vision-based estimation performs better than the LQE from shear stress measurements in any case of poor initialization of the LQE.

The strong robustness to noise of the vision-based state estimation is presented on Fig. 8(c) where the standard deviation (STD) σ_{of} on the optical flow noise has been purposely set to a value 10 times higher than the STD σ_{ss} on the shear stress noise. This figure presents an average over a large number of realizations of the stochastic noises in the case where $N = 501$
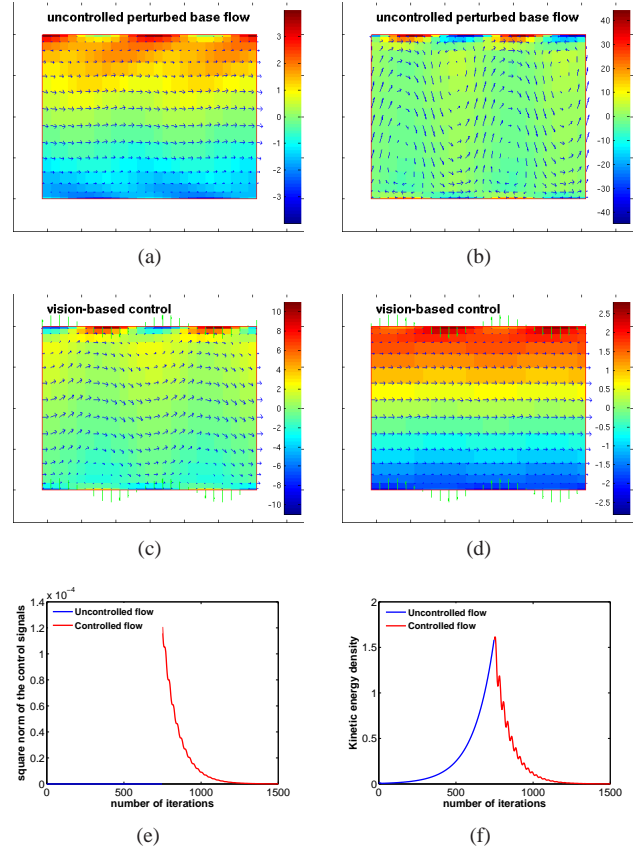

Fig. 7. Vision-based control (the colour in the figures represents the vorticity map): (a) desired image of the flow, (b) initial image of the flow, the perturbation has grown $k = 750$, (c) controlled flow at $k = 1047$, (d) controlled flow at $k = 1500$, (e) control law versus frame iteration and (f) kinetic energy density versus frame iteration.

velocity measurements are used. Note that $N = 501$ is far less than the number of velocity measurements available in real situations where the images size can be at least 1280×960 ($N = 1280$). Due to a large number of flow particles velocities provided by visual sensing, the new approach is very robust to noisy measurements.

3) Behavior of the closed-loop systems: The behaviour of the closed-loop system is shown to be better with the VB-LQG control (30) than with the SSB-LQG control (14). Results are presented in Fig. 9. Fig. 9(a) depicts the behaviour of the control signal in the ideal case (no measurements noise, no initialization error). Fig. 9(b) depicts the behaviour of the control signal when the initial value is set as $\widehat{\mathbf{p^n}}(0) = \mathbf{0}$ by default since the value of $\widehat{\mathbf{p^n}}(0)$ is unknown. In this case we can see that the value of the control signal is 100 times higher than the ideal control signal case which includes the VB-LQG approach (compare the highest control signal values in Fig. 9(b) and 9(a)). This higher control signal value could lead to an unsuitable state trajectory which can cause the real non-linear system to diverge as shown in [22]. In addition, as expected, the control signal (see Fig. 9(b)) takes more time to converge to 0 (3000 iterations compared to the VB-LQG approach). This leads to an energy consumption far much higher for the SSB-LQG control than for the VB-LQG control.

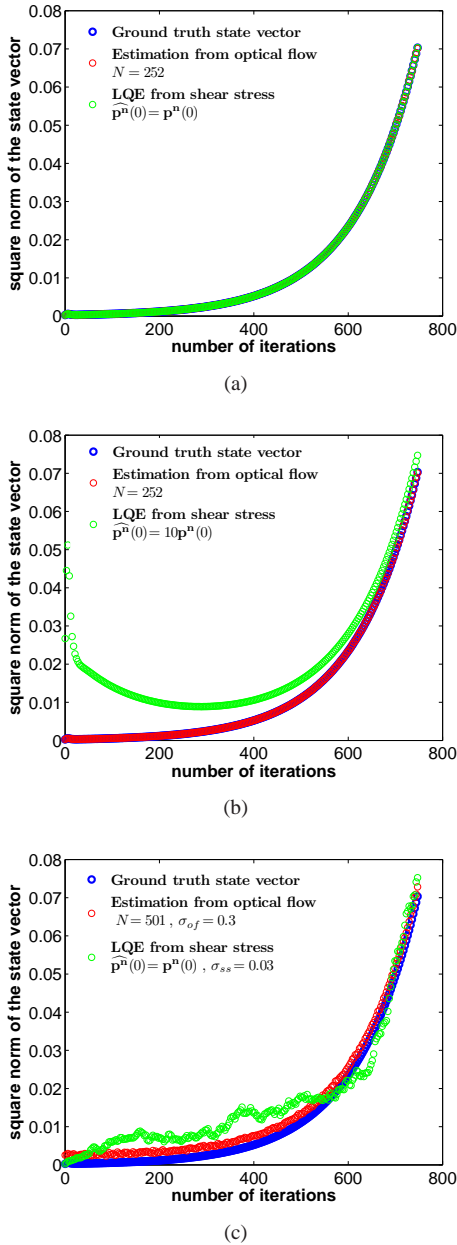The figures in the second row present the control signals in

Fig. 8. Comparison of state vector estimations using shear stress and optical flow: (a) ideal case, (b) LQE poor initialization, (c) measurements noise with a large number of velocities measurements N.
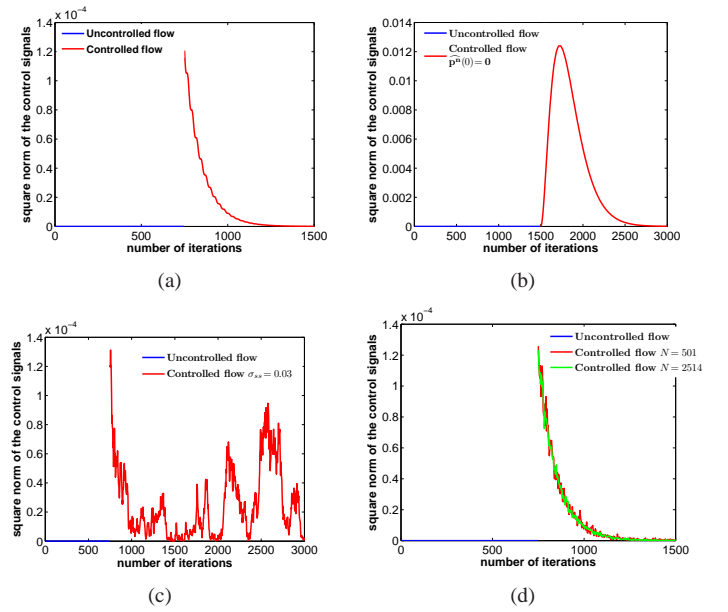

Fig. 9. Comparison SSB-LQG and the VB-LQG control approaches: (a) ideal case (no measurements noise, no initialization error), (b) initialization error in the SSB-LQG control, (c) measurements noise the SSB-LQG control, (d) measurements noise in the VB-LQG control.

B. Reduction of the transient energy growth

In this section we show that the VB-LQG control (30) can be used to limit the transient energy growth better than the SSB-LQG regulator. We have chosen the following classical characteristics as used in related works [33], [16], [28]: the Reynolds number is $R_e = 5000$, the length pair of the channel is $(L_x = 4\pi, L_z = 2\pi)$, and the reduced model Fourier wavenumber pair is $(\alpha_n = 0, \beta_n = 2.044)$. The parameter r of the weighting matrix $\mathbf{R}$ is set as $r = 128$ since it is shown in [22] that this value of r corresponds to the lowest transient energy growth.
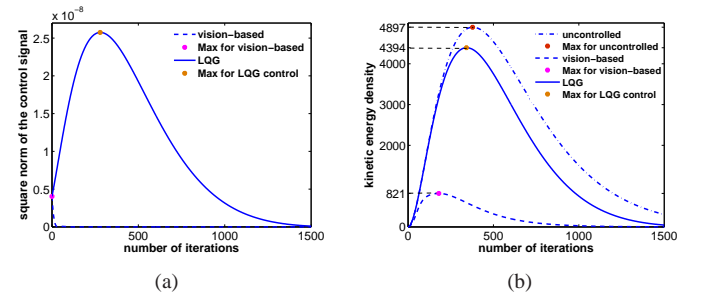
Fig. 10. VB-LQG and SSB-LQG controls for the worst initial conditions: (a) control signals, (b) kinetic energy density.

Let $\mathbf{p^n}_{\text{worst}}(0)$ be the initial condition corresponding the maximum transient energy growth show with the magenta dot on Fig. 4(b) or the magenta dot on the blue dash-dotted plot on Fig. 10(b). In this particular case, if the flow is not controlled the high transient energy could instigate transition to turbulence in the real flow. Here we apply the VB-LQG control (30) and the SSB-LQG control (14) in the ideal case where there is no measurements noise. In addition, as in [22], we assume reasonable to set zero observer initial

presence of measurements noise. Fig. 9(c) pictures the case of the SSB-LQG control where we can see that the control signal does not converge to zero: although the noise STD has been set to a small value, $\sigma_{ss} = 0.03$, the control signal is very noisy, which is not suitable for actuators lifetime. Finally, Fig. 9(d) illustrates the robustness of the VB-LQG control where the STD in the optical flow noise is 10 times higher than the STD in the shear stress noise: we can see from this last figure that the larger the sample of flow particles velocities used the lesser the noise in the control signal.

conditions for the SSB-LQG controller, i.e. $\widehat{\mathbf{p^n}}_{\text{worst}}(0) = \mathbf{0}$. Fig. 10(a) shows the control signals where we can see that the maximum control value for the SSB-LQG is 5 times greater than the maximum control value of the VB-LQG approach. On Fig. 10(b) we can see the kinetic energy is effectively better reduced by the VB-LQG control (30) than by the SSB-LQG control (14). This is mainly due to the asymptotic convergence of the observer used in the SSB-LQG approach. To sum up, the VB-LQG control (30) offers a better reduction of the kinetic energy density with much lesser control efforts than the SSB-LQG control (14).

C. Discussion

In the previous sections we have demonstrated that our vision-based estimation scheme clearly provides better results than the LQE based on sparse measurements of shear stress. Many reasons can explain that.

First, the LQE approach is not well adapted to this problem. Indeed, this approach only provides the asymptotic convergence of the state estimation to the true value of the state. In addition, as shown by equation (34), this approach directly depends on measurements noise through the vector **L**. Moreover, the goal of the LQE method is to provide a state estimation consistent with the linearized model of the flow but not with the true model. A much better approach is to directly extract consistent 2D velocities with a physical model of the flow by methods described in section V-A1b. In this case, by using such methods, the convergence to the true 2D velocities is no longer asymptotic, the robustness with respect to measurements noise is also achieved through an averaging over a dense sample of measurements.

Second, let us assume the availability of dense shear stress measurements. In this case, by averaging over a large number of shear stress measurements we can expect a less noisy estimated state. Nevertheless, such an approach will suffer from a lower spatial resolution than the vision-based approach. Indeed, it seems very difficult to instrument in practice a wall with more shear stress sensors than the number of pixels (in the streamwise direction) of a camera. Therefore, the state estimation from a vision-based approach will still be more robust to noise.

VIII. CONCLUSION

In this paper we have proposed a vision-based approach for fluid flow control. This approach uses image measurements to estimate the flow state. Theoretical proofs have been presented to show the improvements on state estimation and flow control provided by the vision-based approach over the commonly proposed shear stress based LQG control. Indeed the shear stress based LQG regulator limitations concern the limited number of shear stress measurements, the measurements noise and the initialization of the observer involved in the flow state estimation. The initialization issue is not of concerned in the vision-based approach. In addition the vision-based approach has been shown to be robust to measurements noise since a large number of flow velocities is available in real practical situations. These results, validated on a linear simulator, suggest that visual servoing can significantly improve fluid flow control. Future work will be devoted to the validation of the vision-based control approach using a non-linear flow simulator.

APPENDIX A
POISEUILLE FLOW MODELING

The reduced linearized system (6) is obtained from (1) in the following four steps [28].

A. Linearization about the base flow

In the first step, equation (1) is linearized about the steady base flow

$$\begin{cases} \frac{\partial \mathbf{V_b}}{\partial t} + (\mathbf{V_b} \cdot \nabla)\mathbf{V_b} = -\nabla P_b + \frac{1}{R_e}\nabla^2 \mathbf{V_b} \\ \nabla \cdot \mathbf{V_b} = 0 \\ \mathbf{V_b}(x, y = \pm 1, z, t) = \mathbf{0} \end{cases} \quad (40)$$

where $\mathbf{V_b} = (V_{bx}, V_{by}, V_{bz})$ (see (2)). At a given time t_0, a temporal instability is introduced in the base flow (40) through a velocity-pressure perturbation parameter $(\mathbf{V_p}, P_p)$. The resulting velocity-pressure parameter is given by $(\mathbf{V_b} + \mathbf{V_p}, P_b + P_p)$ and the resulting flow motion, obtained from (1) and (40) after some developments, and assumed periodic in the x and z directions is given by

$$\begin{cases} \frac{\partial \mathbf{V_p}}{\partial t} + (\mathbf{V_p} \cdot \nabla)\mathbf{V_p} + (\mathbf{V_p} \cdot \nabla)\mathbf{V_b} + (\mathbf{V_b} \cdot \nabla)\mathbf{V_p} \\ = -\nabla P_p + \frac{1}{R_e}\nabla^2 \mathbf{V_p} \\ \nabla \cdot \mathbf{V_p} = 0 \\ \mathbf{V_p}(x = 0, y, z, t) = \mathbf{V_p}(x = L_x, y, z, t) \\ \mathbf{V_p}(x, y, z = 0, t) = \mathbf{V_p}(x, y, z = L_z, t) \\ \mathbf{V_p}(x, y = \pm 1, z, t) = \mathbf{0}. \end{cases} \quad (41)$$

Assuming that the non-linear term $(\mathbf{V_p} \cdot \nabla)\mathbf{V_p}$ is negligible compared to the other terms of the first equation in (41), we can set $(\mathbf{V_p} \cdot \nabla)\mathbf{V_p} = 0$ in (41), which leads to the linear model of the perturbed flow

$$\frac{\partial V_{px}}{\partial t} + V_{bx}\frac{\partial V_{px}}{\partial x} + V_{py}\frac{dV_{bx}}{dy} = -\frac{\partial P_p}{\partial x} + \frac{1}{R_e}\nabla^2 V_{px} \quad (42a)$$

$$\frac{\partial V_{py}}{\partial t} + V_{bx}\frac{\partial V_{py}}{\partial x} = -\frac{\partial P_p}{\partial y} + \frac{1}{R_e}\nabla^2 V_{py} \quad (42b)$$

$$\frac{\partial V_{pz}}{\partial t} + V_{bx}\frac{\partial V_{pz}}{\partial x} = -\frac{\partial P_p}{\partial z} + \frac{1}{R_e}\nabla^2 V_{pz} \quad (42c)$$

$$\frac{\partial V_{px}}{\partial x} + \frac{\partial V_{py}}{\partial y} + \frac{\partial V_{pz}}{\partial z} = 0 \quad (42d)$$

$$\mathbf{V_p}(x = 0, y, z, t) = \mathbf{V_p}(x = L_x, y, z, t) \quad (42e)$$

$$\mathbf{V_p}(x, y, z = 0, t) = \mathbf{V_p}(x, y, z = L_z, t) \quad (42f)$$

$$\mathbf{V_p}(x, y = \pm 1, z, t) = \mathbf{0}. \quad (42g)$$

B. Divergence-free formulation

The second step consists in a divergence-free formulation of (42). Indeed, since there is no time derivative in the second equation in (42d), equation (42) has to be reformulated into another equation which implicitly takes into account the constraint (42d). The divergence-free formulation can be obtained using the velocity-vorticity approach as done in [33].

Using the velocity-vorticity approach, from (42), we obtain after some developments, the normal-velocity equation

$$
\begin{cases}
(\frac{\partial}{\partial t} + V_{bx}\frac{\partial}{\partial x})\nabla^2 V_{py} - \frac{d^2 V_{bx}}{dy^2}\frac{\partial V_{py}}{\partial x} - \frac{1}{R_e}\nabla^2(\nabla^2 V_{py}) = 0 \\
\hspace{9cm} (43a) \\
V_{py}(x, y = \pm 1, z, t) = 0 \hspace{2.3cm} (43b) \\
\frac{\partial V_{py}(x, y = \pm 1, z, t)}{\partial y} = 0 \hspace{1.9cm} (43c) \\
V_{py}(x = 0, y, z, t) = V_{py}(x = L_x, y, z, t) \hspace{0.3cm} (43d) \\
V_{py}(x, y, z = 0, t) = V_{py}(x, y, z = L_z, t) \hspace{0.3cm} (43e) \\
V_{py}(x, y, z, t_0) = V_{py0}(x, y, z), \hspace{1.3cm} (43f)
\end{cases}
$$

where $V_{py0}(x, y, z)$ represents the initial conditions; and the normal-vorticity equation

$$
\begin{cases}
\frac{\partial \eta_y}{\partial t} + \frac{\partial V_{py}}{\partial z}\frac{dV_{bx}}{dy} + V_{bx}\frac{\partial \eta_y}{\partial x} - \frac{1}{R_e}\nabla^2\eta_y = 0 \hspace{0.2cm} (44a) \\
\eta_y(x, y = \pm 1, z, t) = 0 \hspace{2.7cm} (44b) \\
\eta_y(x = 0, y, z, t) = \eta_y(x = L_x, y, z, t) \hspace{0.5cm} (44c) \\
\eta_y(x, y, z = 0, t) = \eta_y(x, y, z = L_z, t) \hspace{0.5cm} (44d) \\
\eta_y(x, y, z, t_0) = \eta_{y0}(x, y, z) \hspace{1.8cm} (44e)
\end{cases}
$$

where

$$
\eta_y(x, y, z, t) = \frac{\partial V_{px}}{\partial z}(x, y, z, t) - \frac{\partial V_{pz}}{\partial x}(x, y, z, t) \quad (45)
$$

is the vorticity component in the y direction, $\eta_{y0}(x, y, z)$ represents initial conditions. Equation (44) is obtained by simplifying th expression $\partial(42a)/\partial z - \partial(42c)/\partial x$. Equation (43) is obtained in three steps: the Laplacian of (42b) is given by

$$
\frac{\partial \nabla^2 V_{py}}{\partial t} = \quad -V_{bx}\frac{\partial \nabla^2 V_{py}}{\partial x} + \frac{1}{R_e}\nabla^2(\nabla^2 V_{py}) - \frac{\partial \nabla^2 P_p}{\partial y}
$$
$$
-\frac{d^2 V_{bx}}{dy^2}\frac{\partial V_{py}}{\partial x} - 2\frac{dV_{bx}}{dy}\frac{\partial^2 V_{py}}{\partial x \partial y}. \quad (46)
$$

The Laplacian of P_p, obtained by taking the divergence of (42) $(\partial(42a)/\partial x + \partial(42b)/\partial y + \partial(42c)/\partial z)$, is given by

$$
\nabla^2 P_p = \left(\frac{1}{R_e}\nabla^2 - \frac{\partial}{\partial t} - V_{bx}\frac{\partial}{\partial x}\right)\nabla \cdot \mathbf{V_P} - 2\frac{\partial V_{py}}{\partial x}\frac{dV_{bx}}{dy}. \quad (47)
$$

Plugging (42d) into (47) leads to

$$
\nabla^2 P_p = -2\frac{\partial V_{py}}{\partial x}\frac{dV_{bx}}{dy}. \quad (48)
$$

Finally plugging (48) into (46) leads to (43).

C. Closed-loop system equation and system output

In the third step we present the closed-loop system equation and the system output.

1) Closed-loop system equation: In order to ensure zero boundary conditions in the closed-loop controlled system, boundary control inputs are taken into account in the system equations (43) and (44). Since boundary control consists in modifying the upper $y = 1$ and lower $y = -1$ normal velocity, i.e. $V_{py}(x, y = \pm 1, z, t) \neq 0$, it is clear that only equation (43) is concerned with changes in boundary conditions.

The boundary control on the upper and the lower channels can be theoretically modeled by (see Fig. 2)

$$
\begin{cases}
V_{py}(x, y_M, z, t) = \chi_u(x, y_1, z, t) = w_u(x, z, t)f_u(y_1) \\
V_{py}(x, y_1, z, t) = \chi_l(x, y_M, z, t) = w_l(x, z, t)f_l(y_M),
\end{cases} \quad (49)
$$

where $y_M = -1$, $y_1 = 1$; χ_l and χ_u verify Neumann boundary conditions

$$
\frac{\partial \chi_l(x, y_M, z, t)}{\partial y} = 0, \quad \frac{\partial \chi_u(x, y_1, z, t)}{\partial y} = 0; \quad (50)
$$

$w_u(x, z, t)$ and $w_l(x, z, t)$ are sinusoidal functions that verify the mass conservation condition in the closed-system, i.e. the mass of fluid injected by blowing equals the mass of fluid removed by suction; and $f_u(y) = \frac{-y^3 + 3y + 2}{4}$, $f_l(y) = \frac{y^3 - 3y + 2}{4}$ are spatial weighting functions. More precisely the control consists to modify the amplitudes of the sinusoidal functions $w_u(x, z, t)$ and $w_l(x, z, t)$: we thus have a two control inputs system. Note that in the absence of control, i.e. when $w_u(t, x, z) = w_l(t, x, z) = 0$, the red dashed curves (see Fig. 2) are aligned with the lower y_M and upper y_1 boundary lines as expected.

Using the change of variables

$$
\begin{cases}
V_{py}(x, y, z, t) = V_{py}^h(x, y, z, t) + \chi(x, y, z, t) \\
\text{where } \chi(x, y, z, t) = \chi_u(x, y, z, t) + \chi_l(x, y, z, t)
\end{cases} \quad (51)
$$

into (43) leads to the normal-velocity homogeneous formulation of the closed-loop equation

$$
\begin{cases}
\left(\frac{\partial}{\partial t} + V_{bx}\frac{\partial}{\partial x}\right)\nabla^2 V_{py}^h - \frac{d^2 V_{bx}}{dy^2}\frac{\partial V_{py}^h}{\partial x} - \frac{1}{R_e}\nabla^2(\nabla^2 V_{py}^h) \\
= -\left(\frac{\partial}{\partial t} + V_{bx}\frac{\partial}{\partial x}\right)\nabla^2 \chi + \frac{d^2 V_{bx}}{dy^2}\frac{\partial \chi}{\partial x} + \frac{1}{R_e}\nabla^2(\nabla^2 \chi) \\
\hspace{9cm} (52a) \\
V_{py}^h(x, y = \pm 1, z, t) = 0, \quad \frac{\partial V_{py}^h(x, y = \pm 1, z, t)}{\partial y} = 0 \quad (52b) \\
V_{py}^h(x = 0, y, z, t) = V_{py}^h(x = L_x, y, z, t) \hspace{1.5cm} (52c) \\
V_{py}^h(x, y, z = 0, t) = V_{py}^h(x, y, z = L_z, t) \hspace{1.5cm} (52d) \\
V_{py}^h(x, y, z, t_0) = V_{py0}^h(x, y, z). \hspace{2.8cm} (52e)
\end{cases}
$$

2) System output: The flow is usually sensed on its boundary channel by measuring the shear stress at several points on the upper and the lower boundaries. Four non-dimensionalized measurements of the shear stress at the point $(x, z) = (x_i, z_i)$ can be used: the first measurements vector $(z_{xyu}(x_i, y_1, z_i, t), z_{yzu}(x_i, y_1, z_i, t))$ on the upper boundary and the second measurements vector $(z_{xyl}(x_i, y_M, z_i, t), z_{yzl}(x_i, y_M, z_i, t))$ on the lower boundary as done in [22]. Since the computations $z_{xyu}(x_i, y_1, z_i, t)$ and $z_{yzu}(x_i, y_1, z_i, t)$ are similar to the computations of $z_{xyl}(x_i, y_M, z_i, t)$ and $z_{yzl}(x_i, y_M, z_i, t)$ respectively, in

the following we focus only on the computation of $z_{xyl}(x_i, y_M, z_i, t)$ and $z_{yzl}(x_i, y_M, z_i, t)$. We have

$$\begin{cases} z_{xyl}(x_i, y_M, z_i, t) = \dfrac{1}{R_e}\left(\dfrac{\partial V_{px}(x_i, y_M, z_i, t)}{\partial y} + \dfrac{\partial V_{py}(x_i, y_M, z_i, t)}{\partial x}\right) \\[2mm] z_{yzl}(x_i, y_M, z_i, t) = \dfrac{1}{R_e}\left(\dfrac{\partial V_{pz}(x_i, y_M, z_i, t)}{\partial y} + \dfrac{\partial V_{py}(x_i, y_M, z_i, t)}{\partial z}\right) \end{cases}$$
(53)

In the case where no control is applied, we have no slip boundary condition, i.e. $V_{py}(x, y_M, z, t) = 0$ on the lower boundary for instance, thus $\partial V_{py}(x, y_M, z, t)/\partial x = 0$ and $\partial V_{py}(x, y_M, z, t)/\partial z = 0$. On the other hand, in the case where a boundary control is applied, since the value of $V_{py}(x, y_M, z, t)$ is known, the values $\partial V_{py}(x, y_M, z, t)/\partial x$ and $\partial V_{py}(x, y_M, z, t)/\partial z$ are also known. That is why, the shear stress measurement vector can be reduced to [47], [16]:

$$\begin{pmatrix} z_{xyl}(x_i, y_M, z_i, t) \\ z_{yzl}(x_i, y_M, z_i, t) \end{pmatrix} = \begin{pmatrix} \dfrac{1}{R_e}\dfrac{\partial V_{px}(x_i, y_M, z_i, t)}{\partial y} \\[2mm] \dfrac{1}{R_e}\dfrac{\partial V_{pz}(x_i, y_M, z_i, t)}{\partial y} \end{pmatrix}. \quad (54)$$

The expression of the boundary output in terms of the normal velocity and normal vorticity is given in the next section.

D. Spatial discretization via spectral decomposition

Since the flow control problem is an infinite dimension problem in spatial coordinates (i.e. an infinite degrees of freedom system), as a second approximation and last step towards the state space representation (6), the closed-loop linearized equations (52), (44) and the system output (54) are discretized in space in order to solve the problem in practice.

1) Projection in the streamwise and spanwise directions:

a) Approximation of velocities: We recall that in the case of temporal instabilities the flow is assumed periodic in the streamwise and spanwise directions in [47]. Using Fourier series in the streamwise and spanwise directions, the homogeneous normal velocity $V_{py}^h(x, y, z, t)$ can be approximated at a wavenumber pair $(\alpha_n = \alpha_0 n, \beta_n = \beta_0 n)$ with $(\alpha_0 = \frac{2\pi}{L_x}, \beta_0 = \frac{2\pi}{L_z})$ (the fundamental wavenumber pair), by

$$V_{py}^h(x, y, z, t) = 2\text{Real}(V_{py}^{hn}(y, t)e^{i(\alpha_n x + \beta_n z)}), \quad (55)$$

where

$$V_{py}^{hn}(y, t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z} V_{py}^h(x, y, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z. \quad (56)$$

Assuming a similar solution for the streamwise velocity V_{px}, spanwise velocity V_{pz}, normal vorticity η_y, upper boundary control χ_u and lower boundary control χ_l leads respectively to the following approximations at the wavenumber pair (α_n, β_n)

$$V_{px}(x, y, z, t) = 2\text{Real}(V_{px}^n(y, t)e^{i(\alpha_n x + \beta_n z)}), \quad (57)$$

with
$$V_{px}^n(y, t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z} V_{px}(x, y, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z;$$

$$V_{pz}(x, y, z, t) = 2\text{Real}(V_{pz}^n(y, t)e^{i(\alpha_n x + \beta_n z)}), \quad (58)$$

with
$$V_{pz}^n(y, t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z} V_{pz}(x, y, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z;$$

$$\eta_y(x, y, z, t) = 2\text{Real}(\eta_y^n(y, t)e^{i(\alpha_n x + \beta_n z)}), \quad (59)$$

with
$$\eta_y^n(y, t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z}\eta_y(x, y, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z;$$

$$\chi_u(x, y_1, z, t) = 2\text{Real}(f_u(y_1)q_u^n(t)e^{i(\alpha_n x + \beta_n z)}), \quad (60)$$

with

$$q_u^n(t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z} w_u(x, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z \quad (61)$$

and

$$\chi_l(x, y_M, z, t) = 2\text{Real}(f_l(y_M)q_l^n(t)e^{i(\alpha_n x + \beta_n z)}), \quad (62)$$

with

$$q_l^n(t) = \frac{1}{L_x L_z}\int_0^{L_x}\int_0^{L_z} w_l(x, z, t)e^{-i(\alpha_n x + \beta_n z)}\,\mathrm{d}x\,\mathrm{d}z. \quad (63)$$

Finally, the approximation of the normal velocity, obtained from the change of variable equation (51) and from (56), (61) and (63), is given by

$$V_{py}(x, y, z, t) = 2\text{Real}(V_{py}^n(y, t)e^{i(\alpha_n x + \beta_n z)}), \quad (64)$$

where

$$V_{py}^n(y, t) = V_{py}^{hn}(y, t) + f_u(y_1)q_u^n(t) + f_l(y_M)q_l^n(t). \quad (65)$$

b) Approximation of the closed-loop controlled system: Here we present the approximation of the closed-loop system equations (52) and (44) at the wavenumber pair (α_n, β_n).

Using (55), (60) and (62), the approximation of the normal velocity equation (52), obtained after some developments, is given by

$$\begin{cases} \dfrac{\partial}{\partial t}\left(-k_n^2 V_{py}^{hn}(y, t)\dfrac{\partial^2 V_{py}^{hn}(y, t)}{\partial y^2}\right) = -\left(i\alpha_n V_{bx} + \dfrac{2k_n^2}{R_e}\right)\dfrac{\partial^2 V_{py}^{hn}}{\partial y^2} \\[3mm] \quad + \left(i\alpha_n k_n^2 V_{bx} + i\alpha_n\dfrac{d^2 V_{bx}}{dy^2} + \dfrac{k_n^4}{R_e}\right)V_{py}^{hn}(y, t) + \dfrac{1}{R_e}\dfrac{\partial^4 V_{py}^{hn}(y, t)}{\partial y^4} \\[3mm] \quad + \left(i\alpha_n k_n^2 V_{bx} + i\alpha_n\dfrac{d^2 V_{bx}}{dy^2} + \dfrac{k_n^4}{R_e}\right)(f_u(y_1)q_u^n(t) + f_l(y_M)q_l^n(t)) \\[3mm] \quad - \left(i\alpha_n V_{bx} + \dfrac{2k_n^2}{R_e}\right)\left(\dfrac{d^2 f_u(y_1)}{dy^2}q_u^n(t) + \dfrac{d^2 f_l(y_M)}{dy^2}q_l^n(t)\right) \\[3mm] \quad + \left(k_n^2\left(f_u(y_1)\dot{q}_u^n(t) + f_l(y_M)\dot{q}_l^n(t)\right)\right) \\[3mm] \quad - \left(\dfrac{d^2 f_u(y_1)}{dy^2}\dot{q}_u^n(t) + \dfrac{d^2 f_l(y_M)}{dy^2}\dot{q}_l^n(t)\right) & \text{(66a)} \\[3mm] V_{py}^{hn}(y = \pm 1, t) = 0, \quad \dfrac{\partial V_{py}^{hn}(y = \pm 1, t)}{\partial y} = 0 & \text{(66b)} \\[3mm] V_{py}^{hn}(y, t_0) = V_{py0}^{hn}(y), & \text{(66c)} \end{cases}$$

where $k_n^2 = \alpha_n^2 + \beta_n^2$.

Using (59) and (64), after some developments, the normal vorticity equation (44) can be approximated at the wavenumber pair (α_n, β_n) as follows

$$\begin{cases} i\dfrac{\partial \eta_y^n(y, t)}{\partial t} = \left(\alpha_n V_{bx} + \dfrac{k_n^2}{iR_e}\right)\eta_y^n(y, t) - \dfrac{1}{iR_e}\dfrac{\partial^2 \eta_y^n(y, t)}{\partial y^2} \\[3mm] \quad + \beta_n\dfrac{dV_{bx}}{dy}V_{py}^{hn}(y, t) + \beta_n\dfrac{dV_{bx}}{dy}(f_u(y_1)q_u^n(t) + f_l(y_M)q_l^n(t)) \\[3mm] & \text{(67a)} \\[1mm] \eta_y^n(y = \pm 1) = 0 & \text{(67b)} \\[2mm] \eta_y^n(y, t_0) = \eta_{y0}^n(y). & \text{(67c)} \end{cases}$$

c) Approximation of the system output: The approximation at the wavenumber pair (α_n, β_n) of the boundary output (54) in terms of the normal velocity is given by:

$$\begin{cases} z_{xyl}(x_i, y_M, z_i, t) = 2\text{Real}(z_{xyl}^n(y_M, t)e^{i(\alpha_n x_i + \beta_n z_i)}) \\ z_{yzl}(x_i, y_M, z_i, t) = 2\text{Real}(z_{yzl}^n(y_M, t)e^{i(\alpha_n x_i + \beta_n z_i)}) \end{cases}$$
(68)

with

$$\begin{pmatrix} z_{xyl}^n(y_M, t) \\ z_{yzl}^n(y_M, t) \end{pmatrix} = \frac{1}{R_e} \begin{pmatrix} \frac{\partial V_{px}^n(y_M, t)}{\partial y} \\ \frac{\partial V_{pz}^n(y_M, t)}{\partial y} \end{pmatrix}.$$

Now we show how to express $\begin{pmatrix} z_{xyl}^n(y_M, t) \\ z_{yzl}^n(y_M, t) \end{pmatrix}$ using the normal velocity $V_{py}^n(y, t)$ and the normal vorticity $\eta_y^n(y, t)$ Fourier coefficients. From the definition of the normal vorticity given by (45) and from (59), (57) and (58), we easily get the Fourier coefficient $\eta_y^n(y, t)$ in terms of $V_{px}^n(y, t)$ and $V_{pz}^n(y, t)$:

$$\eta_y^n(y, t) = i\left(\beta_n V_{px}^n(y, t) - \alpha_n V_{pz}^n(y, t)\right).$$
(69)

Plugging (57), (64) and (58) into mass conservation equation (42d), leads to

$$-\frac{\partial V_{py}^n(y, t)}{\partial y} = i\alpha_n V_{px}^n(y, t) + i\beta_n V_{pz}^n(y, t).$$
(70)

Using (69) and (70) we easily obtain the Fourier coefficients $V_{px}^n(y, t)$ and $V_{pz}^n(y, t)$ in terms of $\eta_y^n(y, t)$ and $\frac{\partial V_{py}^n(y, t)}{\partial y}$:

$$\begin{pmatrix} V_{px}^n(y, t) \\ V_{pz}^n(y, t) \end{pmatrix} = \frac{i}{k_n^2} \begin{pmatrix} \alpha_n \frac{\partial V_{py}(y, t)}{\partial y} - \beta_n \eta_y^n(y, t) \\ \beta_n \frac{\partial V_{py}(y, t)}{\partial y} + \alpha_n \eta_y^n(y, t) \end{pmatrix}.$$
(71)

From (71) it straightforward to get

$$\begin{pmatrix} z_{xyl}^n(y_M, t) \\ z_{yzl}^n(y_M, t) \end{pmatrix} = \frac{i}{R_e k_n^2} \begin{pmatrix} \alpha_n \frac{\partial^2 V_{py}(y_M, t)}{\partial y^2} - \beta_n \frac{\partial \eta_y^n(y_M, t)}{\partial y} \\ \beta_n \frac{\partial^2 V_{py}(y_M, t)}{\partial y^2} + \alpha_n \frac{\partial \eta_y^n(y_M, t)}{\partial y} \end{pmatrix}.$$
(72)

By plugging (65) into (72), we immediately obtained the approximation at the wavenumber pair (α_n, β_n) of the boundary output given by (68):

$$\begin{cases} z_{xyl}^n(y_M, t) = \frac{i}{k_n^2 R_e} \left(\alpha_n \frac{\partial^2 V_{py}^{hn}(y_M, t)}{\partial y^2} - \beta_n \frac{\partial \eta_y^n(y_M, t)}{\partial y} \right) \\ \qquad + \frac{i}{k_n^2 R_e} \left(\alpha_n q_u^n(t) \frac{d^2 f_u(y_M)}{dy^2} + \alpha_n q_l^n(t) \frac{d^2 f_l(y_M)}{dy^2} \right) \\ z_{yzl}^n(y_M, t) = \frac{i}{k_n^2 R_e} \left(\beta_n \frac{\partial^2 V_{py}(y_M, t)}{\partial y^2} + \alpha_n \frac{\partial \eta_y^n(y_M, t)}{\partial y} \right) \\ \qquad + \frac{i}{k_n^2 R_e} \left(\beta_n q_u^n(t) \frac{d^2 f_u(y_M)}{dy^2} + \beta_n q_l^n(t) \frac{d^2 f_l(y_M)}{dy^2} \right). \end{cases}$$
(73)

2) Decomposition in the normal direction: Now we show how to decompose $V_{py}^{hn}(y, t)$ given in (66) and $\eta_y^n(y, t)$ given in (67) in the normal direction. This decomposition is obtained by evaluation of Chebychev polynomials at Gauss-Lobato collocation points. These points range from $y_M = -1$ to $y_1 = 1$ and are defined as follows

$$y_k = \cos((k - 1)\pi/(M - 1)), \quad 1 \leq k \leq M.$$
(74)

This distribution of points is suitable for spectral accuracy.

On one hand the approximation $\eta_y^n(y, t)$ given in (67) satisfies only Dirichlet as presented in (67b): that is why the decomposition of $\eta_y^n(y, t)$ in the normal direction is done using Chebychev polynomials $(\Theta_m(y))_m$ that satisfied

only Dirichlet boundary conditions. On the other hand the approximation $V_{py}^{hn}(y, t)$ given in (66) satisfies Dirichlet and Neumann boundary conditions as stated in (66b): the decomposition of $V_{py}^{hn}(y, t)$ in the normal direction is thus obtained using Chebychev polynomials $(\Sigma_m(y))_m$ which satisfied both Dirichlet and Neumann boundary conditions. Chebychev polynomials $(\Theta_m(y))_m$ and $(\Sigma_m(y))_m$ can be selected in order to produce the best conditioning of the discretized form of the Laplacian as proven in [22]:

$$\begin{cases} \Theta_{m \geq 3}(y_1) = 0, \ \Theta_{m \geq 3}(y_M) = 0 \\ \Theta_{m \geq 3, \text{odd}}(y) = \Gamma_m(y) - \Gamma_1(y) \\ \Theta_{m \geq 3, \text{even}}(y) = \Gamma_m(y) - \Gamma_2(y) \\ \text{with } \Gamma_m(y_k) = \cos((m - 1)\arccos(y_k)) \end{cases}$$
(75)

and

$$\Sigma_m(y) = (1 - y^2)\Theta_m(y).$$
(76)

The normal vorticity $\eta_y^n(y, t)$ is discretized by evaluation of $(\Theta_m(y))_m$ at Gauss-Lobbato collocation points y_k as follows:

$$\eta_y^n(y_k, t) = \sum_{m=3}^{M} a_{\eta y m}^n(t)\Theta_m(y_k), \ 2 \leq k \leq M - 1. \quad (77)$$

The normal velocity V_{py}^{hn} is discretized by evaluation of $(\Sigma_m(y))_m$ at Gauss-Lobatto collocation points y_k as follows:

$$V_{py}^{hn}(y_k, t) = \sum_{m=3}^{M} a_{vym}^n(t)\Sigma_m(y_k), \ 2 \leq k \leq M - 1. \quad (78)$$

However we have chosen the following definition $(\Sigma_m(y))_m$ also proposed in [22]

$$\begin{cases} \Sigma_{m \geq 5}(y_1) = 0, \ \Sigma_{m \geq 5}(y_M) = 0 \\ \frac{d\Sigma_{m \geq 5}(y = y_1)}{dy} = 0, \ \frac{d\Sigma_{m \geq 5}(y = y_M)}{dy} = 0 \\ \Sigma_{m \geq 5, \text{odd}} = (\Gamma_m - \Gamma_1) - \frac{(m-1)^2}{(m-3)^2}(\Gamma_{m-2} - \Gamma_1) \\ \Sigma_{m \geq 5, \text{even}} = (\Gamma_m - \Gamma_2) - \frac{(m-1)^2 - 1}{(m-3)^2 - 1}(\Gamma_{m-2} - \Gamma_2). \end{cases}$$
(79)

In this case, the normal velocity decomposition is approximated as follows

$$V_{py}^{hn}(y_k, t) = \sum_{m=5}^{M} a_{vym}^n(t)\Sigma_m(y_k), \ 3 \leq k \leq M - 2. \quad (80)$$

3) Reduced Linearized model: Now we present the reduced linearized model which consists of a set of ordinary differential equations (ODE) obtained from the discretization of (66), (67) and (73) in the normal direction.

a) Approximation of the closed-loop controlled system: Plugging (80) into (66) leads to, after some developments, to

$$\begin{cases} \mathbf{A_5^n \dot{a}_{vy}^n}(t) = \mathbf{A_6^n a_{vy}^n}(t) + \mathbf{B_5^n q}(t) + \mathbf{B_6^n \dot{q}}(t) \\ \mathbf{a_{vy}^n}(0) = \mathbf{a_{vy0}^n} \end{cases}$$
(81)

where $\mathbf{a_{vy}^n}(t) = (a_{vym}^n(t))_{1 \leq m \leq M-4}$ is a complex column state vector; $\mathbf{q}^\top(t) = \begin{bmatrix} q_u^{n\top}(t) & q_l^{n\top}(t) \end{bmatrix}^\top$ is the complex inputs vector; $\mathbf{A_5^n} = (a_{5mk}^n)_{1 \leq k, m \leq M-4}$ is a constant real matrix given by $a_{5mk}^n = -k_n^2 \Sigma_{m+4}(y_{k+2}) + \frac{d^2\Sigma_{m+4}(y_{k+2})}{dy^2}$; $\mathbf{A_6^n} = (a_{6mk}^n)_{1 \leq k, m \leq M-4}$ is a constant complex matrix given by

$$a_{6mk}^n = \left(i\alpha_n k_n^2 V_{bx}(y_{k+2}) + i\alpha_n \frac{d^2 V_{bx}(y_{k+2})}{dy^2} + \frac{k_n^4}{R_e}\right)\Sigma_{m+4}(y_{k+2})$$
$$- \left(i\alpha_n V_{bx}(y_{k+2}) + \frac{2k_n^2}{R_e}\right)\frac{d^2\Sigma_{m+4}(y_{k+2})}{dy^2} + \frac{1}{R_e}\frac{d^4\Sigma_{m+4}(y_{k+2})}{dy^4};$$

$\mathbf{B_5^n} = \begin{bmatrix} \mathbf{b_{5u}^n} & \mathbf{b_{5l}^n} \end{bmatrix} \in \mathbb{C}^{(M-4)\times 2}$ is a constant complex matrix given by

$$\mathbf{b_{5u}^n} = (b_{5uk}^n)_{1\leq k\leq M-4}, \ \mathbf{b_{5l}^n} = (b_{5lk}^n)_{1\leq k\leq M-4}$$

and

$$b_{5uk}^n = \left(i\alpha_n k_n^2 V_{bx}(y_{k+2}) + i\alpha_n \frac{d^2 V_{bx}(y_{k+2})}{dy^2} + \frac{k_n^4}{R_e}\right) f_u(y_{k+2})$$
$$- \left(i\alpha_n V_{bx}(y_{k+2}) + \frac{2k_n^2}{R_e}\right) \frac{d^2 f_u(y_{k+2})}{dy^2},$$

$$b_{5lk}^n = \left(i\alpha_n k_n^2 V_{bx}(y_{k+2}) + i\alpha_n \frac{d^2 V_{bx}(y_{k+2})}{dy^2} + \frac{k_n^4}{R_e}\right) f_l(y_{k+2})$$
$$- \left(i\alpha_n V_{bx}(y_{k+2}) + \frac{2k_n^2}{R_e}\right) \frac{d^2 f_l(y_{k+2})}{dy^2};$$

$\mathbf{B_6^n} = \begin{bmatrix} \mathbf{b_{6u}^n} & \mathbf{b_{6l}^n} \end{bmatrix} \in \mathbb{C}^{(M-4)\times 2}$ is a constant complex matrix given by $\mathbf{b_{6u}^n} = \left(k_n^2 f_u(y_{k+2}) - \frac{d^2 f_u(y_{k+2})}{dy^2}\right)_{1\leq k\leq M-4}$

and $\mathbf{b_{6l}^n} = \left(k_n^2 f_l(y_{k+2}) - \frac{d^2 f_l(y_{k+2})}{dy^2}\right)_{1\leq k\leq M-4}$.

Plugging (77) and (80) into (67), we obtain after some developments

$$\begin{cases} \mathbf{A_7^n}\dot{\mathbf{a}}_{\eta\mathbf{y}}^n(t) = \mathbf{A_8^n}\mathbf{a_{vy}^n}(t) + \mathbf{A_9^n}\mathbf{a_{\eta y}^n}(t) + \mathbf{B_7^n}\mathbf{q}(t) \\ \mathbf{a_{\eta y}^n}(0) = \mathbf{a_{\eta y0}^n} \end{cases} \quad (82)$$

where $\mathbf{a_{\eta y}^n}(t) = (a_{\eta y m}^n(t))_{1\leq m\leq M-2}$ is a complex column state vector; $\mathbf{A_7^n} = (a_{7mk}^n)_{1\leq k,m\leq M-2}$ is a constant complex matrix given by $a_{7mk}^n = i\Theta_{m+2}(y_{k+1})$; $\mathbf{A_8^n} = (a_{8km}^n)_{1\leq k\leq M-2, \ 1\leq m\leq M-4}$ is a constant complex matrix given by $a_{8km}^n = \beta_n \frac{dV_{bx}(y_{k+1})}{dy}\Sigma_{m+1}(y_{k+1})$; $\mathbf{A_9^n} = (a_{9mk}^n)_{1\leq k,m\leq M-2}$ is a constant real matrix given by $a_{9mk}^n = (\alpha_n V_{bx}(y_{k+1}) + \frac{k_n^2}{iR_e})\Theta_{m+2}(y_{k+1}) - \frac{1}{iR_e}\frac{d^2\Theta_{m+2}(y_{k+1})}{dy2}$; $\mathbf{B_7^n} = \begin{bmatrix} \mathbf{b_{7u}^n} & \mathbf{b_{7l}^n} \end{bmatrix} \in \mathbb{R}^{(M-2)\times 2}$ is a constant real matrix given by $\mathbf{b_{7u}^n} = \left(\beta_n \frac{dV_{bx}(y_{k+1})}{dy} f_u(y_{k+1})\right)_{1\leq k\leq M-2}$ and $\mathbf{b_{7l}^n} = \left(\beta_n \frac{dV_{bx}(y_{k+1})}{dy} f_l(y_{k+1})\right)_{1\leq k\leq M-2}$.

Using (81) and (82) the state space representation of the 3D controlled flow is given by

$$\begin{cases} \mathbf{A_1^n}\dot{\mathbf{a}}^n(t) = \mathbf{A_2^n}\mathbf{a}^n(t) + \mathbf{B_1^n}\mathbf{q}(t) + \mathbf{B_2^n}\dot{\mathbf{q}}(t) \\ \mathbf{a}^n(0) = \mathbf{a}^n_0 \end{cases} \quad (83)$$

with $\mathbf{a}^n(t) = \begin{pmatrix} \mathbf{a_{vy}^n} \\ \mathbf{a_{\eta y}^n} \end{pmatrix}$, $\mathbf{A_1^n} = \begin{bmatrix} \mathbf{A_5^n} & \mathbf{0} \\ \mathbf{0} & \mathbf{A_7^n} \end{bmatrix}$,

$\mathbf{A_2^n} = \begin{bmatrix} \mathbf{A_6^n} & \mathbf{0} \\ \mathbf{A_8^n} & \mathbf{A_9^n} \end{bmatrix}$, $\mathbf{B_1^n} = \begin{bmatrix} \mathbf{b_{5u}^n} & \mathbf{b_{5l}^n} \\ \mathbf{b_{7u}^n} & \mathbf{b_{7l}^n} \end{bmatrix}$ and $\mathbf{B_2^n} = \begin{bmatrix} \mathbf{b_{6u}^n} & \mathbf{b_{6l}^n} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$.

b) Approximation of the system output: Plugging (77) and (80) into the system output on the lower wall (73) leads, after some developments, to

$$\mathbf{z^n}_l(t) = \mathbf{C_{31l}^{n\top}}\mathbf{a_{vy}^n}(t) + \mathbf{C_{32l}^{n\top}}\mathbf{a_{vy}^n}(t) + \mathbf{D_{1l}^n}\mathbf{q}(t) \quad (84)$$

with $\mathbf{z^n}_l(t) = \begin{bmatrix} \mathbf{z^n}_{xyl}(t) \\ \mathbf{z^n}_{yzl}(t) \end{bmatrix}$,

$\mathbf{C_{31l}^{n\top}} = \begin{bmatrix} \left(\frac{i\alpha_n}{k_n^2 R_e}\frac{d^2\Sigma_{m+4}(y_M)}{dy^2}\right)_{1\leq m\leq M-4} \\ \left(\frac{i\beta_n}{k_n^2 R_e}\frac{d^2\Sigma_{m+4}(y_M)}{dy^2}\right)_{1\leq m\leq M-4} \end{bmatrix}$,

$\mathbf{C_{32l}^{n\top}} = \begin{bmatrix} \left(-\frac{i\beta_n}{k_n^2 R_e}\frac{d\Theta_{m+2}(y_M)}{dy}\right)_{1\leq m\leq M-2} \\ \left(\frac{i\alpha_n}{k_n^2 R_e}\frac{d\Theta_{m+2}(y_M)}{dy}\right)_{1\leq m\leq M-2} \end{bmatrix}$ and

$\mathbf{D_{1l}^n} = \begin{bmatrix} \frac{i\alpha_n}{k_n^2 R_e}\frac{d^2 f_u(y_M)}{dy^2} & \frac{i\alpha_n}{k_n^2 R_e}\frac{d^2 f_l(y_M)}{dy^2} \\ \frac{i\beta_n}{k_n^2 R_e}\frac{d^2 f_u(y_M)}{dy^2} & \frac{i\beta_n}{k_n^2 R_e}\frac{d^2 f_l(y_M)}{dy^2} \end{bmatrix}$.

Similarly we obtain the following discretization of the upper wall output

$$\mathbf{z^n}_u(t) = \mathbf{C_{31u}^{n\top}}\mathbf{a_{vy}^n}(t) + \mathbf{C_{32u}^{n\top}}\mathbf{a_{\eta y}^n}(t) + \mathbf{D_{1u}^n}\mathbf{q}(t) \quad (85)$$

$\mathbf{z^n}_u(t) = \begin{bmatrix} \mathbf{z^n}_{xyu}(t) \\ \mathbf{z^n}_{yzu}(t) \end{bmatrix}$,

$\mathbf{C_{31u}^{n\top}} = \begin{bmatrix} \left(\frac{i\alpha_n}{k_n^2 R_e}\frac{d^2\Sigma_{m+4}(y_1)}{dy^2}\right)_{1\leq m\leq M-4} \\ \left(\frac{i\beta_n}{k_n^2 R_e}\frac{d^2\Sigma_{m+4}(y_1)}{dy^2}\right)_{1\leq m\leq M-4} \end{bmatrix}$,

$\mathbf{C_{32u}^{n\top}} = \begin{bmatrix} \left(-\frac{i\beta_n}{k_n^2 R_e}\frac{d\Theta_{m+2}(y_1)}{dy}\right)_{1\leq m\leq M-2} \\ \left(\frac{i\alpha_n}{k_n^2 R_e}\frac{d\Theta_{m+2}(y_1)}{dy}\right)_{1\leq m\leq M-2} \end{bmatrix}$ and

$\mathbf{D_{1u}^n} = \begin{bmatrix} \frac{i\alpha_n}{k_n^2 R_e}\frac{d^2 f_u(y_1)}{dy^2} & \frac{i\alpha_n}{k_n^2 R_e}\frac{d^2 f_l(y_1)}{dy^2} \\ \frac{i\beta_n}{k_n^2 R_e}\frac{d^2 f_u(y_1)}{dy^2} & \frac{i\beta_n}{k_n^2 R_e}\frac{d^2 f_l(y_1)}{dy^2} \end{bmatrix}$.

Using (84) and (85), the system output is given by

$$\mathbf{z}(t) = \mathbf{C_3^{n\top}}\mathbf{a}^n(t) + \mathbf{D_1^n}\mathbf{q}(t) \quad (86)$$

with $\mathbf{z}(t) = \begin{bmatrix} \mathbf{z^n}_u(t) \\ \mathbf{z^n}_l(t) \end{bmatrix}$, $\mathbf{D_1^n} = \begin{bmatrix} \mathbf{D_{1u}^n} \\ \mathbf{D_{1l}^n} \end{bmatrix}$ and $\mathbf{C_3^{n\top}} = \begin{bmatrix} \mathbf{C_{31u}^{n\top}} & \mathbf{C_{32u}^{n\top}} \\ \mathbf{C_{31l}^{n\top}} & \mathbf{C_{32l}^{n\top}} \end{bmatrix}$.

c) Canonical state space representation: Now we present the canonical state space representation obtained from (83) and (86).

If $\mathbf{A_1^n}$ is invertible, then (83) can be rewritten as

$$\dot{\mathbf{a}}^n(t) = \mathbf{A_3^n}\mathbf{a}^n(t) + \mathbf{B_3^n}\mathbf{q}(t) + \mathbf{B_4^n}\dot{\mathbf{q}}(t), \quad (87)$$

where $\mathbf{A_3^n} = (\mathbf{A_1^n})^{-1}\mathbf{A_2^n}$, $\mathbf{B_3^n} = (\mathbf{A_1^n})^{-1}\mathbf{B_1^n}$ and $\mathbf{B_4^n} = (\mathbf{A_1^n})^{-1}\mathbf{B_2^n}$. Using the augmented state vector

$$\mathbf{p^{n\top}}(t) = \begin{bmatrix} \mathbf{a^{n\top}}(t) & \mathbf{q^\top}(t) \end{bmatrix}^\top, \quad (88)$$

expression (87) and (86) can easily be rewritten in the classical state space representation form as follows

$$\begin{cases} \dot{\mathbf{p}}^n(t) = \mathbf{A^n}\mathbf{p}^n(t) + \mathbf{B^n}\mathbf{u}(t) \\ \mathbf{z}(t) = \mathbf{C^{\top n}}\mathbf{p}^n(t) \\ \mathbf{p}^n(0) = \mathbf{p^n}_0 \end{cases} \quad (89)$$

where

$$\mathbf{A^n} = \begin{bmatrix} \mathbf{A_3^n} & \mathbf{B_3^n} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \ \mathbf{B^n} = \begin{bmatrix} \mathbf{B_4^n} \\ \mathbf{I} \end{bmatrix}, \ \mathbf{u}(t) = \dot{\mathbf{q}}(t), \ \mathbf{C^n} = \begin{bmatrix} \mathbf{C_3^n} \\ \mathbf{D_1^n} \end{bmatrix}.$$

APPENDIX B
VISION-BASED STATE ESTIMATION

A. 2D plane Poiseuille flow

In this case the state vector $\mathbf{p^n}(t)$ equals $(\mathbf{a_{vy}^n}(t), q_u^n(t), q_l^n(t))$ since only (81) is used to derived the reduced model at the wavenumber pair $(\alpha_n = 1, \beta_n = 0)$. Indeed at this wavenumber pair, equations (81) and (82) are decoupled because $\mathbf{A_8^n} = \mathbf{0}$; moreover the control signal $\mathbf{q}(t)$ has no effect in (82) since $\mathbf{B_7^n} = \mathbf{0}$.

We have shown in Appendix A-D that $\mathbf{p^n}(t) = (\mathbf{a_{vy}^n}(t), q_u(t), q_l(t))$ are coefficients of decomposition of the normal velocity $V_{py}(x, y, z, t)$ over both analytical Fourier and Chebychev bases [33]. Note in this case $V_{py}(x, y, z, t) = V_{py}(x, y, t)$ as shown in Section III-A. Here we show the three steps for estimation of $\mathbf{p^n}(t)$ from the perturbation velocity map $\mathbf{V_p}(y, x, t)$ (transpose of $\mathbf{V_p}(x, y, t)$) in the general case where the optical flow measurements are corrupted by a Gaussian noise.

Let

$$\mathbf{V_{py}}(y, x, t) = \begin{bmatrix} V_{py}(y_1, x_1, t) & \cdots & V_{py}(y_1, x_N, t) \\ \cdots & \ddots & \cdots \\ V_{py}(y_M, x_1, t) & \cdots & V_{py}(y_M, x_N, t) \end{bmatrix}$$
(90)

be the ideal normal perturbation velocities, where N and M are the number of velocity measurements of the image of the flow in the streamwise and normal directions respectively. Since optical flow measurements are corrupted by a Gaussian noise, the normal perturbation velocities are also corrupted by a Gaussian noise (see (21)):

$$\widehat{V_{py}}(y_j, x_i, t) = V_{py}(y_j, x_i, t) + \varepsilon_{ji}(t)$$
(91)

where $\varepsilon_{ji}(t)$ is random Gaussian noise on the velocity value at each location (independently) with standard deviation (STD) σ_{of} and 0 mean. Let

$$\boldsymbol{\varepsilon}(t) = \begin{bmatrix} \varepsilon_{11}(t) & \cdots & \varepsilon_{1N}(t) \\ \vdots & \ddots & \vdots \\ \varepsilon_{M1}(t) & \cdots & \varepsilon_{MN}(t) \end{bmatrix}$$
(92)

be the Gaussian noise matrix. In the following we show how to obtain (36).

We first compute the Fourier series coefficients of the measured image velocities $\widehat{\mathbf{V_{py}}}$ at the wavenumber $\alpha_n = 1$. Indeed by multiplying the sum of (90) and (92) by the transpose of the vector

$$\frac{1}{N} \begin{bmatrix} e^{-i\alpha_n x_1} & \cdots & e^{-i\alpha_n x_N} \end{bmatrix},$$

we obtain the Fourier series of the noisy velocities

$$\begin{bmatrix} \widehat{V_{py}^n}(y_1, t) \\ \vdots \\ \widehat{V_{py}^n}(y_M, t) \end{bmatrix} = \begin{bmatrix} V_{py}^n(y_1, t) \\ \vdots \\ V_{py}^n(y_M, t) \end{bmatrix} + \frac{1}{N} \begin{bmatrix} \varepsilon_1^n(t) \\ \vdots \\ \varepsilon_M^n(t) \end{bmatrix}.$$
(93)

Since the upper boundary condition is a sinusoidal function (see (49)), its Fourier series coefficient is given by

$$\widehat{V_{py}^n}(y_M, t) = \widehat{q_u^n}(t) f_u(y_M),$$
(94)

from which we deduce that

$$\widehat{q_u^n}(t) = \frac{1}{f_u(y_M)} V_{py}^n(y_M, t) + \frac{1}{N f_u(y_M)} \varepsilon_M^n(t).$$
(95)

Plugging $q_u^n(t) = \frac{1}{f_u(y_M)} V_{py}^n(y_M, t)$ into (95) leads to

$$\widehat{q_u^n}(t) = q_u^n(t) + \frac{1}{N f_u(y_M)} \varepsilon_M^n(t).$$
(96)

Similarly the coefficient $\widehat{q_l^n}(t)$ for the lower boundary can be expressed as

$$\widehat{q_l^n}(t) = q_l^n(t) + \frac{1}{N f_l(y_1)} \varepsilon_1^n(t).$$
(97)

In the second step we compute the homogeneous coefficients vector $\widehat{\mathbf{V_{py}^{hn}}}$. From the Fourier transform of (51) given in (65), the homogeneous coefficient measurements $\widehat{V_{py}^{hn}}(y_j, t)$ are given by the expression

$$\widehat{V_{py}^{hn}}(y_j, t) = \widehat{V_{py}^n}(y_j, t) - \widehat{q_u^n}(t) f_u(y) - \widehat{q_l^n}(t) f_l(y).$$
(98)

Using (93), (96), (97) and (65), it is easy to rewritte expression (98) as

$$\widehat{V_{py}^{hn}}(y_j, t) = V_{py}^{hn}(y_j, t) + \frac{1}{N} \varepsilon_j^{hn}(t),$$
(99)

where $\varepsilon_j^{hn}(t) = \left(\varepsilon_j^n(t) - \frac{f_u(y_j)}{f_u(y_1)} \varepsilon_1^n(t) - \frac{f_l(y_j)}{f_l(y_M)} \varepsilon_M^n(t) \right)$ and $j = 1, ..., M$. The homogeneous coefficients vector is thus given by

$$\begin{bmatrix} 0 \\ \widehat{V_{py}^{hn}}(y_2, t) \\ \vdots \\ \widehat{V_{py}^{hn}}(y_{M-1}, t) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ V_{py}^{hn}(y_2, t) \\ \vdots \\ V_{py}^{hn}(y_{M-1}, t) \\ 0 \end{bmatrix} + \frac{1}{N} \begin{bmatrix} 0 \\ \varepsilon_2^{hn}(t) \\ \vdots \\ \varepsilon_{M-1}^{hn}(t) \\ 0 \end{bmatrix}.$$
(100)

In the last step, the vector of Fourier series and Chebychev polynomials decomposition vector coeficients $\widehat{\mathbf{a_{vy}^n}}(t)$ is given by

$$\widehat{\mathbf{a_{vy}^n}}(t) = \begin{bmatrix} \widehat{a_{vy5}^n}(t) \\ \vdots \\ \widehat{a_{vyM}^n}(t) \end{bmatrix} = \boldsymbol{\Sigma}^{-1} \begin{bmatrix} \widehat{V_{py}^{hn}}(y_3, t) \\ \vdots \\ \widehat{V_{py}^{hn}}(y_{M-2}, t) \end{bmatrix},$$
(101)

where

$$\boldsymbol{\Sigma} = \begin{bmatrix} \Sigma_5(y_3) & \cdots & \Sigma_M(y_3) \\ \vdots & \ddots & \vdots \\ \Sigma_5(y_{M-2}) & \cdots & \Sigma_M(y_{M-2}) \end{bmatrix}$$

is obtained by evaluation of combination of Chebychev basis on collocation points (79).

Plugging (100) into (101) leads to

$$\widehat{\mathbf{a_{vy}^n}}(t) = \mathbf{a_{vy}^n}(t) + \frac{1}{N} \boldsymbol{\Sigma}^{-1} \begin{bmatrix} \varepsilon_3^{hn}(t) \\ \vdots \\ \varepsilon_{M-2}^{hn}(t) \end{bmatrix},$$
(102)

since $\mathbf{a_{vy}^n}(t) = \boldsymbol{\Sigma}^{-1} \begin{bmatrix} V_{py}^{hn}(y_3, t) \\ \vdots \\ V_{py}^{hn}(y_{M-2}, t) \end{bmatrix}$. Finally by letting

$$\mathbf{p^n}(t) = \begin{bmatrix} \mathbf{a^n}(t) \\ q_u^n(t) \\ q_l^n(t) \end{bmatrix} \text{ and } \mathbf{e^n}(t) = \begin{bmatrix} \boldsymbol{\Sigma}^{-1} \begin{bmatrix} \varepsilon_3^{hn}(t) \\ \vdots \\ \varepsilon_{M-2}^{hn}(t) \end{bmatrix} \\ \frac{1}{f_u(y_1)} \varepsilon_1^n(t) \\ \frac{1}{f_l(y_M)} \varepsilon_M^n(t) \end{bmatrix},$$

it is easy to express the estimated coefficient $\widehat{\mathbf{p^n}}(t)$ in terms of the ideal one and the noise as given in (36).

B. 3D plane Poiseuille flow

Suppose that the 3D velocities $\mathbf{V_p} = (V_{py}, V_{py}, V_{pz})$ of the flow are measured in a $N \times M \times N_z$ size volume using a stereo-PIV technique with no image measurements noise. Here we show how to compute the state vector $\mathbf{p^n}(t)$ from $\mathbf{V_p}(x,y,z,t)$. The state vector $\mathbf{p^n}(t)$ includes not only $q_u^n(t)$ and $q_l^n(t)$ but also coefficients $\mathbf{a_{vy}^n}(t)$ and $\mathbf{a_{\eta y}^n}(t)$ (see Appendix A-D3a) of decomposition of the normal velocity $V_{py}(x,y,z,t)$ and normal vorticity $\eta_y(x,y,z,t) = \frac{\partial V_{px}(x,y,z,t)}{\partial z} - \frac{\partial V_{pz}(x,y,z,t)}{\partial x}$ respectively, over both analytical Fourier and Chebyshev bases [33]. The state vector $\mathbf{p^n}(t)$ is thus obtained by projecting both the normal perturbation velocity $V_{py}(x,y,z,t)$ and the normal vorticy $\eta_y(x,y,z,t)$ onto Fourier and Chebychev bases as shown below.

Coefficients $\mathbf{a_{vy}^n}(t)$, $q_u^n(t)$ and $q_l^n(t)$ are obtained exactly as in the 2D case (see Appendix B-A), except that here we suppose no measurements noise and the Fourier series coefficients are given by

$$V_{py}^n(y_k, t) = \frac{1}{NN_z} \sum_{k_x=1}^{N} \sum_{k_z=1}^{N_z} V_{py}(y_k, x_{k_x}, z_{k_z}) e^{-i(\alpha_n x_{k_x} + \beta_n z_{k_z})},$$
(103)

with $k = 1, .., M$.

The coefficients $\mathbf{a_{\eta y}^n}(t)$ are obtained by decomposition of the Fourier series coefficients of $\eta_y(x,y,z)$ on Chebychev polynomials evaluated on collocation points y_k [22]:

$$\mathbf{a_{\eta y}^n}(t) = \mathbf{\Theta}^{-1} \begin{bmatrix} \eta_y^n(y_2, t) \\ \vdots \\ \eta_y^n(y_{M-1}, t) \end{bmatrix}$$
(104)

where $\mathbf{\Theta} = \begin{bmatrix} \Theta_3(y_2) & \cdots & \Theta_M(y_2) \\ \vdots & \ddots & \vdots \\ \Theta_3(y_{M-1}) & \cdots & \Theta_M(y_{M-1}) \end{bmatrix}$ and $\eta_y(y_k, t) = i(\beta_n V_{px}^n(y_k, t) - \alpha_n V_{py}^n(y_k, t))$ with

$$V_{px}^n(y_k, t) = \frac{1}{NN_z} \sum_{k_x=1}^{N} \sum_{k_z=1}^{N_z} V_{px}(y_k, x_{k_x}, z_{k_z}) e^{-i(\alpha_n x_{k_x} + \beta_n z_{k_z})},$$

$$V_{pz}^n(y_k, t) = \frac{1}{NN_z} \sum_{k_x=1}^{N} \sum_{k_z=1}^{N_z} V_{pz}(y_k, x_{k_x}, z_{k_z}) e^{-i(\alpha_n x_{k_x} + \beta_n z_{k_z})}.$$

REFERENCES

[1] A. Flaig, "Eco-efficient by design," in *Challenges for aerodynamics engineers for future aircraft design, 8th World Congress on Computational Mechanics*, Venice, Italy, 2008.

[2] D. W. Bechert and M. Bartenwerfer, "The viscous flow on surfaces with longitudinal ribs," *J. of Fluid Mech.*, vol. 206, pp. 105–129, 1989.

[3] H. Choi, P. Moin, and J. Kim, "Direct numerical simulation of turbulent flow over riblets," *J. Fluid Mech.*, vol. 255, pp. 503–539, 1993.

[4] C.-H. Bruneau and I. Mortazavi, "Passive control of the flow around a square cylinder using porous media," *Int. J. for Numerical Methods in Fluids*, vol. 46, no. 4, pp. 415–433, 2004.

[5] J. Favier, A. Dauptain, D. Basso, and A. Bottaro, "Passive separation control using a self-adapted hairy coating," *J. Fluid Mech.*, vol. 627, pp. 451–483, 2009.

[6] I. Wygnanski, "Boundary layer and flow control by periodic addition of momentum," in *4th AIAA Shear Flow Control Conference*, Snowmass Village, Colorado, 1997.

[7] B. Protas and J. Weisfred, "Drag force in the open-loop control of the cylinder wake in the laminar regime," *Physics of Fluids*, vol. 14, no. 2, pp. 810–826, 2002.

[8] M. Bergmann and L. Cordier, "Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models," *Journal of Computational Physics*, vol. 227, no. 16, pp. 7813–7840, 2008.

[9] T. Bewley, "Flow control: new challenges for a new rennaissance," *Progress in Aerospace Sciences*, vol. 37, pp. 21–58, 2001.

[10] A. Seifert, A. Darabi, and I. Wygnanski, "Delay of airfoil stall by periodic excitation," *J. of Aircraft*, vol. 33, no. 4, pp. 691–699, 1996.

[11] D. Greenblatt and I. Wygnanski, "The control of flow separation by periodic excitation," *Prog. Aero. Sci*, vol. 36, no. 7, pp. 487–545, 2000.

[12] A. Brunn and W. Nitsche, "Active control of turbulent separated flows over slanted surfaces," *Int. Journal of Heat and Fluid Flow*, vol. 27, pp. 748–755, 2006.

[13] M. Rouméas, P. Gilliéron, and A. Kourta, "Analysis and control of the near-wake flow over a square-back geometry," *Computers & Fluids*, vol. 38, no. 1, pp. 60–70, 2009.

[14] B. Protas and A. Styczek, "Optimal rotary control of the cylinder wake in the laminar regime," *Physics of Fluids*, vol. 14, no. 7, pp. 2073–2087, Jul 2002.

[15] S. Joshi, J. Speyer, and J. Kim, "A systems theory approach to the feedback stabilization on infinitesimal and finite amplitude disturbances in plane Poiseuille flow," *J. Fluid Mech.*, vol. 332, pp. 157–184, 1997.

[16] T. Bewley and S. Liu, "Optimal and robust control and estimation of linear paths to transition," *J. Fluid Mech.*, vol. 365, pp. 305–349, 1998.

[17] M. Pastoor, R. King, B. Noack, and G. Tadmor, "Observers and feedback control for shear layer vortices," in *44th IEEE Conf. on Decision and Control, and the European Control Conf.*, Seville, Spain, pp. 506–511.

[18] J. Naughton and M. Sheplak, "Modern developments in shear stress measurement," *Progress in Aerospace Sciences*, vol. 38, pp. 515–570, Jul. 2002.

[19] L. Cattafesta and M. Sheplak, *Fundamentals and Applications of Modern Flow Control*, ser. AIAA Progress in Astronautics and Aeronautics, Dec 2009, vol. 231, ch. Actuators and Sensors, pp. 149–175.

[20] S. Joshi, J. Speyer, and J. Kim, "Finite dimensional optimal control of Poiseuille flow," *Journal of Guidance, Control and Dynamics*, vol. 22, pp. 340–348, 1999.

[21] J. Hœpffner, M. Chevalier, T. Bewley, and D. Henningson, "State estimation in wall-bounded flow systems. part 1. perturbed laminar flows," *J. Fluid Mech.*, vol. 534, pp. 263–294, 2005.

[22] J. McKernan, "Control of plane Poiseuille flow: a theoretical and computational investigation," Ph.D. dissertation, Cranfield University, 2006.

[23] J. Kim and T. Bewley, "A linear systems approach to flow control," *Annu. Rev. Fluid Mech.*, vol. 39, pp. 383–417, 2007.

[24] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr, "A multigrid platform for real-time motion computation with discontinuity-preserving variational methods," *Int. J. of Computer Vision*, vol. 70, no. 3, pp. 257–277, Dec. 2006.

[25] F. Champagnat, A. Plyer, G. Le Besnerais, L. B., and Y. Le Sant, "How to calculate dense piv vector fields at video rate," in *8th Int. Symp. On Particle Image Velocimetry*, Melbourne, Victoria (Australia), Aug. 2009.

[26] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: a survey," *Journal of Intelligent & Robotic Systems*, vol. 53, no. 3, pp. 263–296, 2008.

[27] F. Chaumette and S. Hutchinson, "Visual servoing and visual tracking," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, ch. 24, pp. 563–583.

[28] J. McKernan, G. Papadakis, and J. Whidborne, "A linear state-space representation of plane Poiseuille flow for control design - a tutorial," *Int. Journal of Modeling, Identification and Control*, vol. 1, no. 4, pp. 272–280, 2006.

[29] O. Aamo and T. Fossen, "Tutorial on feedback control of flows, part I: stabilization of fluid flows in channels and pipes," *Modeling, Identification and Control*, vol. 23, no. 3, pp. 161–226, 2002.

[30] A. Gleyer and M. Amitay, "Synthetic jets," *Annual Review of Fluid Mechanics*, vol. 34, pp. 503–529, 2002.

[31] L. Cattafesta and M. Sheplak, "Actuators for active flow control," *Annual Review of Fluid Mechanics*, vol. 43, no. 1, pp. 247–272, 2011.

[32] S. Orszag, "Accurate solution of the Orr-Sommerfeld stability equation," *J. Fluid Mech.*, vol. 50, pp. 689–703, 1971.

[33] K. Butler and B. Farrell, "Three-dimensional optimal perturbations in viscous shear flow," *Physics of Fluids A*, vol. 4, no. 8, pp. 1637–1650, 1992.

[34] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, June 1992.

[35] C. Samson, M. Le Borgne, and B. Espiau, *Robot control: the Task Function approach.* Oxford University Press, 1991.

[36] D. Heitz, E. Mémin, and C. Schnörr, "Variational fluid flow measurements from image sequences: synopsis and perspectives," *Exp. Fluids*, vol. 48, no. 3, pp. 369–393, 2010.

[37] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[38] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323–344, Aug 1987.

[39] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of Imaging Understanding Workshop*, 1981, pp. 121–130.

[40] J. Liu and L. Shen, "Fluid flow and optical flow," *J. Fluid Mech.*, vol. 614, pp. 253–291, 2008.

[41] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[42] T. Corpetti, E. Mémin, and P. Pérez, "Dense estimation of fluid flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 365–380, March 2002.

[43] T. Corpetti, D. Heitz, G. Arroyo, E. Mémin, and A. Santa Cruz, "Fluid experimental flow estimation based on an optical-flow scheme," *Exp. Fluids*, vol. 40, no. 1, pp. 80–97, 2006.

[44] M. Raffel, C. Willert, and J. Kompenhans, *Particle image velocimetry: a practical guide*. Springer, 1998.

[45] T. Hori and J. Sakakibara, "High-speed scanning stereoscopic PIV for 3D vorticity measurement in liquids," *Meas. Sci. Technol.*, vol. 15, pp. 1067–1078, 2004.

[46] B. Ganapathisubramani and E. Longmire, "Dual-plane PIV technique to determine the complete velocity gradient tensor in a turbulent boundary layer," *Exp. Fluids*, vol. 39, pp. 222–231, 2005.

[47] S. Joshi, "A system theory approach to the control of plane Poiseuille flow," Ph.D. dissertation, University of California, LA, 1996.