

Onofrio Semeraro · Jan O. Pralits

Full-order optimal compensators for flow control: the multiple inputs case

Received: 29 January 2017 / Accepted: 13 March 2018
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

Abstract Flow control has been the subject of numerous experimental and theoretical works. We analyze full-order, optimal controllers for large dynamical systems in the presence of multiple actuators and sensors. The full-order controllers do not require any preliminary model reduction or low-order approximation: this feature allows us to assess the optimal performance of an actuated flow without relying on any estimation process or further hypothesis on the disturbances. We start from the original technique proposed by Bewley et al. (*Meccanica* 51(12):2997–3014, 2016. <https://doi.org/10.1007/s11012-016-0547-3>), the *adjoint of the direct-adjoint* (ADA) algorithm. The algorithm is iterative and allows bypassing the solution of the algebraic Riccati equation associated with the optimal control problem, typically infeasible for large systems. In this numerical work, we extend the ADA iteration into a more general framework that includes the design of controllers with multiple, coupled inputs and robust controllers (\mathcal{H}_∞ methods). First, we demonstrate our results by showing the analytical equivalence between the full Riccati solutions and the ADA approximations in the multiple inputs case. In the second part of the article, we analyze the performance of the algorithm in terms of convergence of the solution, by comparing it with analogous techniques. We find an excellent scalability with the number of inputs (actuators), making the method a viable way for full-order control design in complex settings. Finally, the applicability of the algorithm to fluid mechanics problems is shown using the linearized Kuramoto–Sivashinsky equation and the Kármán vortex street past a two-dimensional cylinder.

Keywords Flow control · Optimization · Stability analysis

1 Introduction

Linear control has been applied to a large variety of flows in the last decades [2–5]. The interest in such applications has been invigorated by the possible, numerous outcomes ranging from drag reduction to acoustic emission mitigation. From a physical point of view, one of the main assumptions is that the flow can be modified by properly acting on the small-amplitude perturbations before the transition to turbulence occurs or on the large coherent structures at higher Reynolds numbers. If the linear approximation is valid, rigorous methods for linear control can be applied.

Communicated by Daniel J. Bodony.

O. Semeraro (✉)
Laboratoire d’informatique pour la mécanique et les sciences de l’ingénieur (LIMSI), CNRS, Orsay, France
E-mail: onofriosem@gmail.com

J. O. Pralits
Department of Civil, Chemical and Environmental Engineering (DICCA), University of Genoa, Genoa, Italy
E-mail: jan.pralits@unige.it

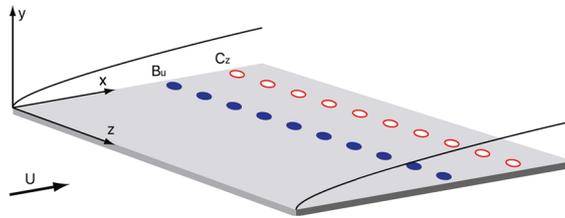


Fig. 1 Sketch of a control unit for boundary layer flows; the blue dots indicate a row of localized actuators (B_u), the red circles are the sensors (C_z) for the performance evaluation. The aim is to modify the flow using actuators in order to fulfill a control objective. Adapted from [4] (color figure online)

In this paper we focus on active control, which by definition is characterized by energy input into the system. Control energy is introduced by means of actuators, and the system to be controlled is referred to as the *plant*. The presence of actuators allows having more flexibility in the control design, while sensors can be introduced such that the actuation is optimized with respect to an objective function.

An example of a plant based on a fluid mechanics system is given in Fig. 1, where the sketch of a boundary layer flow developing on a flat plate is shown, including m actuators B_u and p sensors C_z . We seek a control law $u(t)$, feeding the m actuators. A possible way is to define a *control kernel* $K \in \mathbb{R}^{m \times n}$ such that the control signal $u(t)$ is proportional to the state vector $q \in \mathbb{R}^n$, i.e., $u(t) = Kq$. The controller is designed to fulfill a target; in linear quadratic regulators (LQR), this target is represented by the quadratic function

$$\mathcal{J} = \frac{1}{2} \int_0^T (z^H z + u^H R u) dt \quad (1)$$

to be minimized. The first term includes the signals recorded by the sensors C_z , as $z = C_z v$. The matrix R contains the control penalties for tuning the control effort (see for instance [6] and reference therein).

A difficulty in fluid mechanics is represented by the dimensions of the dynamical systems: although it is relatively common to deal with numerical simulations with a number of degrees of freedom $n > 10^6$, control design tools become infeasible for much smaller dimensions. A standard approach to circumvent this limitation consists in replacing the full-order system with a reduced-order model capturing the *essential* dynamics of the system. This methodology is called *reduce-then-design* and, besides the modeling step, is necessary to enable the application of the control in realistic cases thanks to the reduced number of degrees of freedom.

1.1 Full-order control design

Control design is characterized by numerous choices; among the others, we can mention the location and the spatial distribution of the sensors/actuators pairs, their number, or the coupling among these pairs. For this reason, given a plant, it is not always straightforward to assess whether or not a system is controllable and—when possible—if the performances of the closed-loop are the best achievable. During this preliminary step, full-order controllers can be of help in guiding the design of the control unit as we avoid the model reduction step or the estimation design. However, the full-order controllers cannot be directly used in real application: a reduction in the control dimensionality is still required; moreover, we are limiting ourselves to an optimization perspective, without facing directly the robustness issue [5].

With this in mind, an optimal controller K can be obtained as solution of the continuous algebraic Riccati equation (CARE) [7], for systems of dimensions $n \approx 10^3 - 10^4$ [8]. Alternative methods for the full-order design in large systems were proposed in the last decades, based on the solution of surrogate systems of equations, iterative procedures or algorithms exploiting the inherent sparsity of the plants. A common trait of these methods is their feasibility when the number of actuators $m \ll n$. A classic technique is the Chandrasekhar method: the full-order CARE is replaced by a set of partial differential equations. Alternatively, the projection on low-order Krylov spaces by means of Arnoldi-type processes (see [9] and references therein) and Newton methods have been proposed [10]. An efficient solution method for the Chandrasekhar equations was proposed in [11], where long-time integrators are used in combination with reduced-order models based on proper orthogonal decomposition (POD) for the control of the Kármán vortex street developing behind a cylinder.

Full-order controllers were computed for the control of the channel flow by [12]; in this work, optimal control kernels with a compact support in physical space are designed by solving an optimal control problem

for each wavenumber pair in Fourier space, independently. Indeed, in the limit of parallel flows, the optimal control problems in Fourier space are decoupled from each other as suggested by [13]. An inverse Fourier transform allows reconstructing the kernel in physical domain. The procedure was applied to weakly non-parallel, spatially developing flows [14, 15], although the main drawback is the distribution of sensors/actuators pairs that cannot be localized in space using this approach.

Methods based on the pole placement include the minimum-energy control (MCE), discussed in [1]; in this limit, the control kernel is computed using the unstable adjoint modes of the system. Examples from fluid mechanics are given by [16] for the control of vortex shedding behind a cylinder, and [17] for the control of the instabilities rising in the wake of a thick plate at higher Reynolds number. In the latter contribution, the linearization is performed around different mean flows, computed as RANS and U-RANS solutions.

1.2 Iterative methods

Finally, iterative methods can be considered. The standard approach consists of an iteration based on the direct equation describing the system dynamics and its adjoint (see [18] and citations therein). The unknown of the iteration is the optimal control law $u(t)$ in a given time window; the approach can be generalized to nonlinear settings or to the design of controllers with receding time-horizon of optimization ([19–21]).

However, the control signal in the standard direct-adjoint iteration (hereafter indicated with DA) depends on the initial condition from which the trajectory emanates. This limitation is tackled in Bewley et al. [1], where the *adjoint of the direct-adjoint* (ADA) algorithm is introduced, a method capable of identifying the gain K . From the theoretical point of view, the ADA algorithm replaces the original direct-adjoint optimization with the corresponding sensitivity analysis by considering the adjoint of the entire problem. This procedure changes an optimization problem of size n , the number of states, into a problem of size m , the number of inputs. In [22], the ADA technique is applied to the control of Tollmien–Schlichting waves developing in a two-dimensional boundary layer flow and extended to the estimation problem.

1.3 Present investigation and organization of the work

Building on the findings by [1], in this work we further analyze the algorithm by extending it into a more generalized framework that includes the design of multiple inputs controllers and a class of robust controllers. The optimal control problem is stated in Sect. 2. The ADA algorithm is described in Sect. 3, where the original formulation is revisited and an analytical demonstration applied to a scalar problem is introduced. In Sect. 4, we discuss the design of multiple inputs controllers, optimal estimators and robust, optimal controllers \mathcal{H}_∞ [23]. We compare the multiple inputs ADA algorithm with an iterative methodology based on the stochastic gradient introduced in [24] (Sect. 5). The comparison makes use of a simple distributed system. Finally, in Sect. 6, we apply the algorithm to fluid mechanics applications: the two-dimensional version of the Kuramoto–Sivashinsky (KS) equation in Sect. 6.1 and the flow past a two-dimensional cylinder in Sect. 6.2. The work finalizes with conclusions in Sect. 7.

2 Linear quadratic regulator (LQR): statement of the problem, classical approaches and multiple-variable systems

In this section, we concisely state the optimal control problem and derive the time-continuous algebraic Riccati equation (CARE). For a deeper discussion, we refer to the specialized literature [7]. We introduce the input–output equations

$$\frac{dv}{dt} = Av + B_u u, \quad \text{with } v(0) = v_0, \quad (2a)$$

$$z = C_z v, \quad (2b)$$

where $A \in \mathbb{R}^{n \times n}$ is the system matrix and the variable n indicates the degrees of freedom. In this work, we consider time-continuous, spatially discretized systems. The spatial distribution of m actuators is described by the matrix $B_u \in \mathbb{R}^{n \times m}$. The variable z is scalar and represents a time signal. If the system arises from the

discretization of the Navier–Stokes equations linearized around a baseflow or a mean flow, the vector $\mathbf{v}(t) \in \mathbb{R}^n$ represents the state of the fluid system. We want to identify a control signal $\mathbf{u}(t)$ such that

$$\mathcal{J} = \frac{1}{2} \int_0^T \left(\mathbf{v}^H \mathbf{W} \mathbf{v} + \mathbf{u}^H \mathbf{R} \mathbf{u} \right) dt + \frac{1}{2} \mathbf{v}(T)^H \mathbf{W}_T \mathbf{v}(T) \quad (3)$$

is minimized. The matrices $\mathbf{W} \geq 0 \in \mathbb{R}^{n \times n}$, $\mathbf{R} > 0 \in \mathbb{R}^{m \times m}$ and $\mathbf{W}_T \geq 0 \in \mathbb{R}^{n \times n}$ contain weights in the entries. In our setup, the matrix \mathbf{W} is defined as $\mathbf{W} = \mathbf{C}_z \mathbf{C}_z^H$ and is low-rank (see Fig. 1 and Eq. 1); moreover, the matrix \mathbf{R} is assumed to be diagonal, and the final condition \mathbf{W}_T is set to be null. These assumptions do not lead to any loss of generality.

The solution of the control problem in Eqs. (2–3) can be obtained by defining an augmented cost function $\tilde{\mathcal{J}}$

$$\tilde{\mathcal{J}} = \mathcal{J} - \int_0^T \boldsymbol{\lambda}^H \left(\frac{d\mathbf{v}}{dt} - \mathbf{A} \mathbf{v} - \mathbf{B}_u \mathbf{u} \right) dt. \quad (4)$$

Applying integration by parts, the following system of equations is cast

$$\frac{d\mathbf{v}}{dt} = \mathbf{A} \mathbf{v} + \mathbf{B}_u \mathbf{u}, \quad \text{with } \mathbf{v}(0) = \mathbf{v}_0, \quad (5a)$$

$$\frac{d\boldsymbol{\lambda}}{dt} = -\mathbf{A}^H \boldsymbol{\lambda} - \mathbf{W} \mathbf{v}, \quad \text{with } \boldsymbol{\lambda}(T) = 0, \quad (5b)$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \mathbf{u}} = \mathbf{B}_u^H \boldsymbol{\lambda} + \mathbf{R} \mathbf{u}. \quad (5c)$$

The equation for the adjoint state $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ is obtained by zeroing the gradient $\partial \tilde{\mathcal{J}} / \partial \mathbf{v}$ and is integrated backward in the interval $t \in [T, 0]$. The matrix \mathbf{A}^H denotes the adjoint operator, satisfying the inner-product $\langle \mathbf{A} \mathbf{v}, \boldsymbol{\lambda} \rangle = \langle \mathbf{v}, \mathbf{A}^H \boldsymbol{\lambda} \rangle$. The unknown of the system is the control signal $\mathbf{u}(t)$ in the time interval $t \in [0, T]$. The solution can be approximated using iterations: the control signal is updated at each step of the direct-adjoint iteration as

$$\mathbf{u}(t)^{i+1} = \mathbf{u}(t)^i - \beta^i \left(\frac{\partial \tilde{\mathcal{J}}}{\partial \mathbf{u}} \right)^i, \quad (6)$$

using Eq. 5(c). A gradient descent algorithm can be used for defining the step β [25]. The resulting control signal is optimal for a given initial condition \mathbf{v}_0 . In what follows, we will refer to this technique as direct-adjoint (DA) iteration.

The system in Eq. 5 can be written in compact form as $\dot{\mathbf{x}} = \mathbf{T} \mathbf{x}$ where the vector \mathbf{x} contains the direct state \mathbf{v} and the adjoint state $\boldsymbol{\lambda}$. The matrix $\mathbf{T} \in \mathbb{R}^{2n \times 2n}$ is Hamiltonian and defined as

$$\mathbf{T} = \begin{bmatrix} \mathbf{A} & -\mathbf{B}_u \mathbf{R}^{-1} \mathbf{B}_u^H \\ -\mathbf{W} & -\mathbf{A}^H \end{bmatrix}.$$

In the following, the upper-right block of the Hamiltonian matrix $\mathbf{B}_u \mathbf{R}^{-1} \mathbf{B}_u^H$ —containing the spatial distribution of the actuators \mathbf{B}_u —will be denoted as $\boldsymbol{\Psi}$.

A direct solution of the control problem is obtained by solving the associated Riccati equation. Assuming the relation $\boldsymbol{\lambda} = \mathbf{X} \mathbf{v}$, the algebraic Riccati equation for the control

$$\mathbf{A}^H \mathbf{X} + \mathbf{X} \mathbf{A} - \mathbf{X} \boldsymbol{\Psi} \mathbf{X} + \mathbf{W} = \dot{\mathbf{X}} \quad (7)$$

is obtained. The equation can be solved, for instance, by involving Runge–Kutta time-integrations. Also, for linear time-invariant system in the steady case limit $\dot{\mathbf{X}} = 0$, a direct solution may be computed by taking an ordered Schur decomposition of the Hamiltonian matrix \mathbf{T} (Laub’s method, see [26]). The resulting solution is the matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$, positive-definite and symmetric. The control signal $\mathbf{u}(t)$ is proportional to the state \mathbf{v} as $\mathbf{u}(t) = \mathbf{K} \mathbf{v}$; the constant control kernel $\mathbf{K} \in \mathbb{R}^{m \times n}$ is finally computed as

$$\mathbf{K} = -\mathbf{R}^{-1} \mathbf{B}_u^H \mathbf{X}. \quad (8)$$

For large-system $n > 10^3$ the direct solution of the Riccati equation is not feasible, due to computational costs of order $O(n^3)$ —regardless of the structure of the system matrix \mathbf{A} —and storage requirements which are at least of order $O(n^2)$, see [9]. A viable alternative is represented by iterative methods for the computation of the control kernel \mathbf{K} .

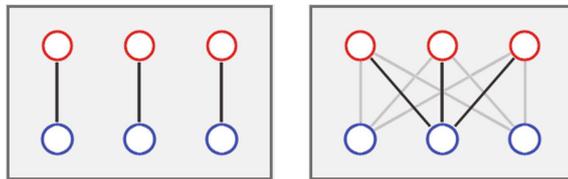


Fig. 2 Two different control-units with the same number of actuators (blue circles) and performance sensors (red circles) are sketched. On the left, each actuator is “connected” with one sensor, leading to a control-unit composed by three single-input single-output (SISO) controllers. In this case, the control-unit is based on a *decentralized design*. On the right, one controller is designed based on all elements. This design accounts for the cross talks among the actuators and is usually referred to as *centralized design*, or multiple inputs multiple-outputs (MIMO) design (color figure online)

2.1 Multiple-variable systems: the MIMO design

In the previous section, we tacitly discussed the plant to be controlled by considering m actuators \mathbf{B}_U . Yet, we did not mention the implications on the control design when multiple actuators are introduced in the plant. Indeed from the design point of view, a multiple-variable system is potentially characterized by cross-couplings between the inputs (actuators) and the outputs (sensors for the objective function or *performance* sensors). These cross-couplings are one of the difficulties in multiple-variable control [27]: indeed, a change in one input can affect multiple-outputs (and vice versa).

Within the context of flow control, this is typically the case of full, three-dimensional configurations. An example is sketched in Fig. 1, based on past investigations discussed in [6] or [28]. When only one actuator/sensor pair is considered, we usually refer to a single-input single-output (SISO) system. The opposite case is when multiple inputs/outputs are introduced; in this case, we refer to multiple inputs multiple-outputs (MIMO) system. For sake of clarity, we consider the cartoon in Fig. 2. A first, simplistic approach for the MIMO design consists of designing a number of SISO closed-loop that equals the number of sensor/actuator pairs; considering the example in Fig. 2, $m = p$ closed-loop controllers are designed, one for each of the actuator-sensor couples, located at the same position z along the spanwise direction. This approach is called *decentralized* approach. In this framework, the number of actuators must equal the number of sensors and the cross-couplings are disregarded, despite this choice may affect both the stability and the performance of the closed-loop system; indeed, the stability of the closed-loop systems is not guaranteed. However, if the decentralized controller is stable in each SISO loop and weak couplings characterize the original plant, then the closed-loop can be also stable. In terms of performance, the resulting controller will be sub-optimal. The opposite approach consists of designing a controller based on one control unit including all the sensors and actuators. This methodology is called *centralized control* design. This choice guarantees optimal performance and stability of the closed-loop, although it is less easy to implement in practical situations.

In conclusion, we can summarize as follows.

1. *Decentralized control*: m control gains \mathbf{K} of dimension $1 \times n$ are designed independently from each other, solving m Riccati equations.
2. *Centralized control*: one control gain \mathbf{K} of dimensions $m \times n$ is designed, by solving one Riccati equation.

In the following, we will discuss only these two limits. Hybrid solutions are possible, where only a part of the sensors is connected with some actuators according to their interactions (*pairing problem*).

3 The adjoint of the direct-adjoint (ADA) algorithm for the solution of the LQR problem

The DA iteration does not allow the direct computation of the optimal control kernel $\mathbf{K} \in \mathbb{R}^n$; indeed, the unknown of the problem for a given initial condition \mathbf{v}_0^i is the control signal $\mathbf{u}(t)^i$ defined in $t \in [0, T]$, with T the final time of optimization. However, as observed by [1], the following linear system can be formed

$$[\mathbf{u}_0^1 \ \mathbf{u}_0^2 \ \dots \ \mathbf{u}_0^n] = \mathbf{K}_{(1 \times n)} [\mathbf{v}_0^1 \ \mathbf{v}_0^2 \ \dots \ \mathbf{v}_0^n]_{(n \times n)}, \quad (9)$$

where n different solutions $\mathbf{u}(t)^i$ of the optimal control problem emanating from n linearly independent initial conditions \mathbf{v}_0^i are used. Thus, n iterations need to be solved, each with a different initial condition. In particular, the known vector on the left-hand side can be formed by taking the corresponding control signal at $t = 0$. The

columns of the matrix on the right-hand side are represented by the initial conditions of each of the n iterative loops. The last step consists of the solution of a linear system of dimensions $n \times n$; so, for large systems this iterative scheme is again infeasible.

However, as elucidated in [1], one may drastically reduce the computational costs of the problem by analyzing the sensitivity with respect to the initial condition using the *adjoint* of the DA system. The original demonstration of the algorithm makes use of integration by parts and it is reported in ‘‘Appendix B’’ for sake of completeness. Here, we propose an alternative version based on the properties of the Hamiltonian systems. Introducing the symplectic matrix

$$J = \begin{bmatrix} \mathbf{0} & I \\ -I & \mathbf{0} \end{bmatrix},$$

the following property is fulfilled

$$T^H J + J T = \mathbf{0}. \quad (10)$$

Let us introduce the state $\mathbf{x} = \begin{pmatrix} v \\ \lambda \end{pmatrix}$ as solution of the system $\dot{\mathbf{x}} = T\mathbf{x}$, with initial condition $\mathbf{x}_0 = \begin{pmatrix} v_0 \\ \lambda_T \end{pmatrix}$ and its *adjoint* state $\tilde{\mathbf{x}} = \begin{pmatrix} \tilde{v} \\ \tilde{\lambda} \end{pmatrix}$, solution of the system $\dot{\tilde{\mathbf{x}}} = T^H \tilde{\mathbf{x}}$, with initial condition $\tilde{\mathbf{x}}_0 = \begin{pmatrix} \tilde{v}_0 \\ \tilde{\lambda}_T \end{pmatrix}$. Note that the state $\tilde{\mathbf{x}}$ is the adjoint of the direct state \mathbf{x} with respect to the symplectic product $\Omega(t)$, defined as

$$\Omega(t) = \tilde{\mathbf{x}}(t)^H J \mathbf{x}(t) = \tilde{\lambda}(t)^H v(t) - \tilde{v}(t)^H \lambda(t). \quad (11)$$

In Hamiltonian systems, this product is constant $\forall t$; including the boundary conditions, we obtain $\Omega(t) = 0$. The relation 11 can be compared with the optimality condition at $t = 0$

$$\tilde{\lambda}_0^H v_0 = \tilde{v}_0^H \lambda_0, \quad (12a)$$

$$\mathbf{u} = \mathbf{K} v_0 = (-\mathbf{R}^{-1} \mathbf{B}_u^H) \lambda_0. \quad (12b)$$

Considering the case $m = 1$, if we introduce the row vector $-\mathbf{R}^{-1} \mathbf{B}_u^H$ as initial condition of the dual system \tilde{v}_0 , the adjoint solution $\tilde{\lambda}_0^H$ corresponds to \mathbf{K} . This is the exact solution of the optimal problem for $T \rightarrow \infty$, obtained without solving the algebraic equation Eq. 7; in practice, a sufficiently long-time window for the optimization guarantees convergence toward this solution.

3.1 An analytical demonstration in 1D

In this section, an analytical demonstration of the ADA algorithm is provided considering a scalar system $n = 1$ with $m > 1$ inputs. The demonstration will be useful for the discussion of the MIMO systems in Sect. 4.

We consider the following scalar system

$$\frac{d}{dt} \begin{pmatrix} v \\ \lambda \end{pmatrix} = \begin{bmatrix} a & -\psi \\ -w & -a \end{bmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix}, \quad (13)$$

with m scalar inputs, such that $\mathbf{b} = (b_1, b_2, \dots, b_m)$; the term ψ reads

$$\psi = (b_1, b_2, \dots, b_m) \mathbf{R} \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{pmatrix} = \sum_{i=1}^m \frac{b_i^2}{r_i}. \quad (14)$$

Note that the last equivalence is true only if \mathbf{R} is a diagonal matrix. The Riccati equation for such a system reduces to a second order equation, whose solutions are

$$x = \frac{a}{\psi} \pm \frac{1}{\psi} \sqrt{a^2 + w\psi}. \quad (15)$$

The solutions x are scalar-valued and $\alpha = \sqrt{a^2 + w\psi}$. Taking the positive solution, the resulting Kálmán gains for the control of the system in Eq. 13 are

$$k_i = -\frac{b_i}{r_i} \left[\frac{a + \alpha}{\psi} \right], \quad (16)$$

with $i = 1, 2, \dots, m$.

It can be shown that the analytical solution of the system in (13) leads to the same result when the ADA algorithm is applied with $T \rightarrow \infty$. The so-called *unit solutions* technique is applied; the method is described in the books by [7, 29] and summarized in ‘‘Appendix A’’. Here, we limit ourselves to observe that a general solution of the system in Eq. 13 can be found by combining two initial-value problems with

1. initial condition $v(t_0) = v_0, \lambda(t_0) = 0$,
2. initial condition $v(t_0) = 0, \lambda(t_0) = 1$.

The constant c is defined using boundary conditions at $t = 0$ and $t = T$. The resulting solutions of the system are

$$v(t) = \frac{(\alpha - a)v_0 + c\psi}{2\alpha}e^{-\alpha t} + \frac{(\alpha + a)v_0 - c\psi}{2\alpha}e^{\alpha t}, \quad (17)$$

$$\lambda(t) = \frac{qv_0 + c(\alpha + a)}{2\alpha}e^{-\alpha t} + \frac{-qv_0 + c(\alpha - a)}{2\alpha}e^{\alpha t}. \quad (18)$$

We now introduce the ADA solution. In particular, we take

$$v(0) = \frac{b_i}{r_i}, \quad \text{with } i = 1, 2, \dots, m \quad (19a)$$

$$\lambda(T) = 0, \quad (19b)$$

$v(T)$ free and $\lambda(0) = k_i$. For $\alpha T \rightarrow \infty$, the terminal conditions in Eq. 19 applied to the solution in Eq. 18, lead to the following relation

$$\lambda(0) = k_i = v(0) \frac{w}{\alpha - a} = -\frac{b_i}{r_i} \left(\frac{\alpha + a}{\psi} \right),$$

that corresponds to Eq. 16. By observing that the Riccati solution in 1D reads

$$x = \frac{\alpha + a}{\psi},$$

the demonstration shows that the ADA machinery replaces the Riccati solution \mathbf{X} ; similarly to matrix-free methods based on the time-steppers, the Riccati solution is never written but its action is approximated numerically using the iterative scheme [30]. This result can be applied also to higher-order systems [29].¹

4 Multiple-variable version of ADA and other extensions

As shown in Sect. 3, the ADA algorithm does not identify a low-rank representation for the matrix \mathbf{X} , but approximates its action using a direct-adjoint iterative scheme. In the presence of m actuators, the design of the controller \mathbf{K} needs to be performed row-by-row: the design of a control kernel of dimensions $\mathbf{K} \in \mathbb{R}^{m \times n}$ requires m iteration loops. Due to this requirement, a question naturally arises: how can we guarantee the coupling among the actuator/sensor pairs?

The answer is found by observing the off-diagonal blocks of the matrix governing the Hamiltonian system, whose solution is approximated by the ADA iteration. The cross-coupling among the actuators is imposed by choosing the term Ψ during the design of the input–output system, while the coupling of the sensor is done selecting $\mathbf{W} = \mathbf{C}_z^H \mathbf{C}_z$. Indeed, in the Riccati problem, the i -th row of the control gain is computed as $\mathbf{K}_i = -\mathbf{R}_{(i,i)}^{-1} \mathbf{B}_{u,i}^H \mathbf{X}$: the coupling is guaranteed by \mathbf{X} , that is approximated here by means of the iterative scheme. Thus, the initial condition of each of the iterations is chosen by taking $\tilde{v}_0^H = -\mathbf{R}_{(i,i)}^{-1} \mathbf{B}_{u,i}^H$, such that each of the corresponding i -th rows of the matrix \mathbf{K} is obtained as an adjoint solution of the iteration process.

In conclusion, we can design a centralized controller by running m iterative processes with Ψ and \mathbf{W} containing all the actuators and sensors—respectively—and by only changing the initial condition. However, note that by choosing a-priori the pairs of sensors/actuators that share the strongest interactions, we can adequately design Ψ and \mathbf{W} without including all the pairs. In the limit where only one actuator is considered for each of the m iterations, we recover the decentralized case where one controller for each actuator is designed independently from the other elements.

¹ A real solution for $n > 1$ can be found in [7, p. 156].

4.1 Estimation problem

The technique can be extended to any problem based on the solution of algebraic Riccati equations. A first example is provided by the estimation problem as shown in [22] for the single-input single-output setting. An estimator is defined by the following dynamical system

$$\dot{\hat{\mathbf{v}}} = \mathbf{A}\hat{\mathbf{v}} + \mathbf{B}_u\mathbf{u} - \mathbf{L}(\mathbf{y} - \hat{\mathbf{y}}), \quad (20a)$$

$$\hat{\mathbf{y}} = \mathbf{C}_y\hat{\mathbf{v}}. \quad (20b)$$

The estimator allows reconstructing the original state \mathbf{v} , based on local measurements \mathbf{y} ; the state $\hat{\mathbf{v}}$ is defined such that the error $\mathbf{y} - \hat{\mathbf{y}}$ is minimized. The estimator is driven by the error term via the matrix \mathbf{L} , referred to as estimation gain. The estimation gain \mathbf{L} is the unknown of the associated estimation problem, obtained as solution of the following Riccati equation

$$\mathbf{A}\mathbf{Y} + \mathbf{Y}\mathbf{A}^H - \mathbf{Y}\mathbf{C}_y^H\mathbf{G}^{-1}\mathbf{C}_y\mathbf{Y} + \mathbf{B}_d\mathbf{W}\mathbf{B}_d^H = 0, \quad (21a)$$

$$\mathbf{L} = -\mathbf{Y}\mathbf{C}_y^H\mathbf{G}^{-1}. \quad (21b)$$

The \mathbf{G} contains in the diagonal entries the estimation penalty, similarly to the control case. From the mathematical point of view, a fictitious adjoint problem is cast such that the cost function

$$\mathcal{N}(\boldsymbol{\lambda}(\tilde{\mathbf{y}}), \tilde{\mathbf{y}}) = \frac{1}{2} \int_0^T (\boldsymbol{\lambda}^H \mathbf{W} \mathbf{p} + \tilde{\mathbf{y}}^H \mathbf{G} \tilde{\mathbf{y}}) dt \quad (22)$$

is minimized. In this dual system, the output \mathbf{C}_y^H is an input of the system, while the adjoint inputs \mathbf{B}_d^H and \mathbf{B}_u^H play the role of the outputs [31]. The analogy is completed by observing that the “feedback law” is now represented by $\tilde{\mathbf{y}}(t) = \mathbf{L}^H \boldsymbol{\lambda}(t)$. Thus, the first term of the cost function expresses the energy of the dual state $\boldsymbol{\lambda}$, while the second term minimizes the energy input of the feedback law.

A justification of this deterministic approach is given by [3]. The physical interpretation of the method is given within the stochastic framework, where the unknown of the Riccati equation (21) is interpreted as the expected energy of the estimation error $\mathbf{e}(t)$, while \mathbf{W} is the covariance of the disturbance forcing [32].

The optimization machinery employed for the control problem can be adopted in an analogous manner for computing the full-dimensional estimation gain \mathbf{L} . The full formulation is given in [22], where the algorithm is referred to as *Adjoint of the Adjoint-Direct* (AAD). Also the AAD algorithm can be generalized to the centralized/decentralized case. Thus, in the presence of p estimation sensors \mathbf{C}_y , p estimators can be designed based only on one sensor, independently from each other, or one centralized estimators can be designed by considering the system with all the p sensors \mathbf{C}_y .

4.2 Riccati solution for \mathcal{H}_∞ problem

We complete this section of extensions of the ADA algorithm considering the \mathcal{H}_∞ problem. The optimal control framework can be extended to the solution of robust optimization problems introducing the *worst* disturbance scenario. The cost function associated with the problem is

$$\mathcal{J} = \frac{1}{2} \int_0^T (\mathbf{v}^H \mathbf{W} \mathbf{v} + \mathbf{u}^H \mathbf{R} \mathbf{u} - \gamma^{-2} \mathbf{d}^H \mathbf{W} \mathbf{d}) dt, \quad (23)$$

where the extra-term to maximize is the unknown disturbance \mathbf{d} . The process consists of a simultaneous optimization problem: the signal \mathbf{u} is computed such that the worst-case disturbance is minimized (*min-max* optimization). The control is sub-optimal as the parameter γ is chosen by the user. For $\gamma \rightarrow \infty$, we approach the limit of the LQR problem already considered. The direct equation of the problem reads

$$\frac{d\mathbf{v}}{dt} = \mathbf{A}\mathbf{v} + \mathbf{B}_d\mathbf{d} + \mathbf{B}_u\mathbf{u}, \quad \text{with } \mathbf{v}(0) = \mathbf{v}_0. \quad (24)$$

By introducing the corresponding augmented Lagrangian, it is possible to define the adjoint equation

$$\frac{d\boldsymbol{\lambda}}{dt} = -\mathbf{A}^H \boldsymbol{\lambda} - \mathbf{W}\mathbf{v}, \quad \text{with } \boldsymbol{\lambda}(T) = 0. \quad (25)$$

Two optimal conditions are defined, by zeroing the respective gradients. The first condition corresponds to the optimal control signal \mathbf{u} ,

$$\mathbf{u} = -\mathbf{R}^{-1} \mathbf{B}_u^H \mathbf{p}, \quad (26)$$

the second condition is the worst disturbance \mathbf{d}

$$\mathbf{d} = \gamma^2 \mathbf{W}^{-1} \mathbf{B}_d^H \mathbf{p}. \quad (27)$$

Imposing the equivalence $\mathbf{p} = \mathbf{X}\mathbf{q}$, into the Hamiltonian system

$$\mathbf{T} = \begin{bmatrix} \mathbf{A} & \gamma^2 \mathbf{B}_d \mathbf{W}^{-1} \mathbf{B}_d^H - \mathbf{B}_u \mathbf{R}^{-1} \mathbf{B}_u^H \\ -\mathbf{W} & -\mathbf{A}^H \end{bmatrix},$$

a CARE equation is obtained also for this problem as

$$\mathbf{A}^H \mathbf{X} + \mathbf{X} \mathbf{A} - \mathbf{X} \left(\mathbf{B}_u \mathbf{R}^{-1} \mathbf{B}_u^H - \gamma^2 \mathbf{B}_d \mathbf{W}^{-1} \mathbf{B}_d^H \right) \mathbf{X} + \mathbf{Q} = 0, \quad (28a)$$

$$\mathbf{K} = -\mathbf{R}^{-1} \mathbf{B}_u^H \mathbf{X}, \quad (28b)$$

$$\mathbf{Y} = \gamma^2 \mathbf{W}^{-1} \mathbf{B}_d^H \mathbf{X}. \quad (28c)$$

The control gain \mathbf{K} and the worst disturbance \mathbf{Y} are computed based on the solution \mathbf{X} of the CARE equation. Thus, the ADA algorithm can be applied for the full-order approximation of this problem. Considering the single-input single-output setting, \mathbf{K} and \mathbf{Y} are computed as adjoint solutions at \mathbf{T} by introducing as the initial conditions

$$\mathbf{q}_{\mathbf{K},0} = -\mathbf{R}^{-1} \mathbf{B}_u^H, \quad (29a)$$

$$\mathbf{q}_{\mathbf{Y},0} = \gamma^2 \mathbf{W}^{-1} \mathbf{B}_d^H, \quad (29b)$$

respectively. In this framework, the robust problem is the solution of a multiple inputs Riccati equation where all the inputs are coupled together. Of course, a MIMO controller with more than one disturbance \mathbf{B}_d and $m > 1$ actuators \mathbf{B}_u can be also designed.

From the numerical point of view, a relevant difference between the standard \mathcal{H}_2 approach and the \mathcal{H}_∞ problem is the optimization process; in the \mathcal{H}_∞ case, we seek a saddle point of the objective function corresponding to the values that simultaneously minimize and maximize it. This optimization requires more specialized techniques that we will consider in future investigations (see for instance [33,34]). In the next sections, we focus on the ADA algorithm only considering full-order LQR controllers.

5 Comparison with other iterative algorithms

In this section, we assess the performance of the ADA algorithm in the presence of multiple inputs and centralized control. We consider a toy-problem and compare ADA with a recent algorithm proposed by Mårtensson and Rantzer in [35]. The test bed is mainly meant at verifying the speed of the convergence for the different algorithms. Fluid mechanics applications are discussed in Sect. 6.

5.1 Full-order controllers using stochastic gradients

The algorithm introduced in [35], and here indicated as MR, is used for comparison. The algorithm computes the control gain by updating at each step the solution as

$$\mathbf{K}^{i+1} = \mathbf{K}^i - \beta^i \left(\nabla_{\mathbf{K}} \tilde{\mathcal{J}} \right)^i, \quad (30)$$

where $\nabla_{\mathbf{K}} \tilde{\mathcal{J}}$ is the gradient of \mathcal{J} with respect to \mathbf{K} . The basic formulation is proposed in ‘‘Appendix C’’. In principle, the algorithm suffers from the same limitations of the DA iteration: the identified solution depends on the initial condition. The problem is circumvented by choosing a new initial condition randomly at each iteration. The method can be interpreted as a stochastic gradient descent; at each step of the iteration, an initial state is chosen for which the gradient is computed. Using different initial states at every iteration, the final solution will not depend on any of the initial conditions.

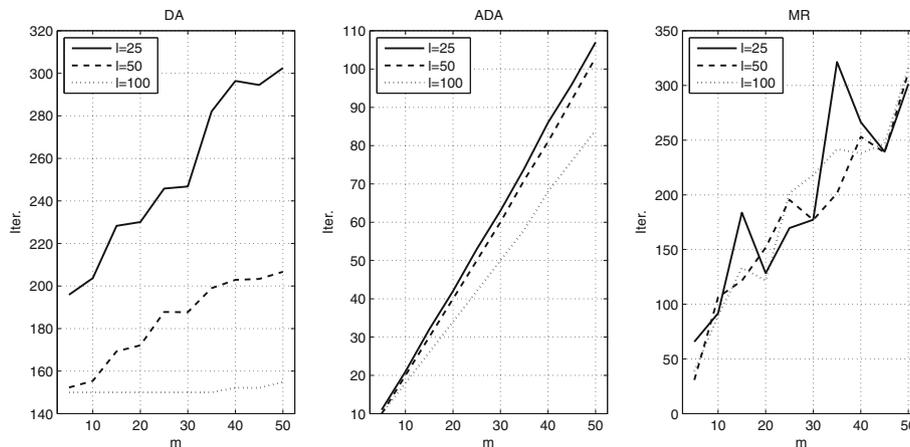


Fig. 3 Convergence of the iterative algorithms. The number of iterations as a function of the number of actuators m and control penalty l is considered for the algorithms described in Sects. 2, 3 and 5.3

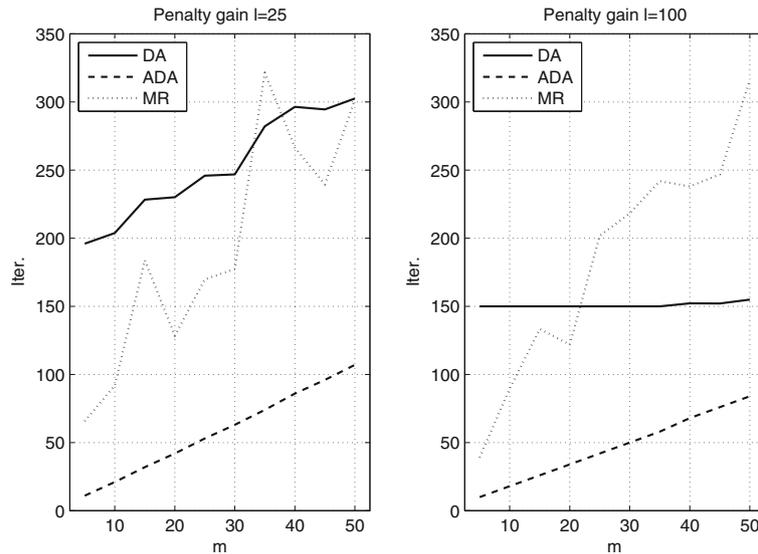


Fig. 4 Comparison among the iterative algorithms for $l = 25$ and $l = 100$. The ADA algorithm is characterized by high scalability with the number of actuators and a number of iterations smaller with respect to the alternative choices

system, together with the storage of a large matrix containing along the columns the n initial conditions; in that sense, the use of the DA algorithm is infeasible for large systems and is considered here only for the sake of completeness.

The MR algorithm has a less clear behavior, due to the choice of the initial conditions; an increase in the number of iterations is observed with m also for this case. The choice of the penalty does not affect the number of iterations required. This is illustrated in Fig. 4, where the number of iterations as a function of m is shown for $l = 25$ and $l = 100$.

We can summarize as follows the main features observed for the three algorithms

1. For all the cases, an increase in the total number of iterations with m is observed. This is expected in ADA, where m iteration loops are required, while it is less obvious for the MR case, where only one iteration loop is required regardless of m .
2. ADA scales linearly with m . This is not the case for the MR algorithm, due to the stochastic gradient application.
3. For the MR algorithm, the averaged number of iterations is not affected by the control effort applied (i.e., the choice of the control penalty l).

In general, we found that the ADA algorithm performs considerably better and is more robust if compared with the analogous iterative algorithm proposed by [35].

6 Numerical examples

In this section, we report two fluid mechanics examples: a planar flow governed by the linear Kuramoto–Sivashinsky equation (Sect. 6.1) and the flow past a two-dimensional cylinder (Sect. 6.2).

6.1 Kuramoto–Sivashinsky model

A modified version of the Kuramoto–Sivashinsky equation is used for testing the multiple inputs case, mimicking the three-dimensional setup sketched in Fig. 1. The flow field is two-dimensional, defined in the x - z plane, and governed by the equation

$$\frac{\partial v}{\partial t} = -V \frac{\partial}{\partial x} \left(v - \frac{1}{8P} \frac{\partial^2 v}{\partial z^2} \right) - \frac{1}{R} \left(P \frac{\partial^2 v}{\partial x^2} + \frac{\partial^4 v}{\partial x^4} + S \frac{\partial^4 v}{\partial z^4} \right). \quad (31)$$

With respect to the original equation, the dynamics is linearized around the convective velocity, $V = 0.4$. The non-dimensional terms R and P are defined as

$$R = \frac{VL^3}{\mu}, \quad P = \frac{\eta L^2}{\mu}, \quad (32)$$

with η representing an energy production term and μ the dissipation; L is the reference length of the system. The parameters are chosen ad-hoc such that the dispersion relation of the system mimics the evolution of traveling packets of Tollmien–Schlichting (TS) waves on a flat plate at $Re_{\delta^*} = 1000$ (see [6]). Introducing the maximum streamwise wavenumber α_{\max} , the maximum spanwise wavenumber β_{\max} and the complex temporal frequency ω , the non-dimensional numbers can be expressed as

$$R = \frac{P^2}{4\omega_{\max}}, \quad P = \frac{2}{\alpha_{\max}^2}, \quad S = \frac{\omega_{\max} R}{\beta_{\max}^4}, \quad (33)$$

with S providing the modulation along the spanwise direction of the wave.

The computational box extends along the streamwise direction in the interval $x \in [0, 500]$ and along the spanwise direction in $z \in [-90, 90]$; a grid with $N_x = 256$ and $N_z = 96$ is chosen. The spatial discretization is performed by means of a pseudo-spectral method that includes a fringe region extending between $x = 400$ and $x = 500$; periodic boundary conditions are imposed along the spanwise direction. The time marching is performed using a three-steps Runge–Kutta scheme; the basic implementation can be found in the repository <https://github.com/nfabbiane/ks2D>. More details are reported by [36].

6.1.1 A multiple inputs full-order controller

The dynamics of the waves governed by the linear KS are depicted in Fig. 5a, c, e, where the impulse response of the system is shown at three different instants. The disturbance is introduced at $\mathbf{x}_0 = (x_0, z_0) = (2.5, 0)$, and its spatial distribution is modeled as

$$\mathbf{B} = \exp \left(-\frac{(x - x_0)^2}{\sigma^2} - \frac{(z - z_0)^2}{\sigma^2} \right), \quad (34)$$

with $\sigma = 4$. The dynamics of the wavepackets mimics the evolution of a TS wave, growing as it propagates downstream along the streamwise direction and extending along the spanwise direction with a backward bending. The corresponding energy growth associated with the evolving wavepacket is shown in Fig. 6b (black, solid line); the root mean square energy is obtained by averaging over $T = 10,000$ time units a stochastically driven linear simulation.

A LQR controller is designed using the ADA algorithm. The setup closely resembles the one analyzed by [6] and reproduced schematically in Fig. 1: a row composed by $n = 9$ localized actuators (\mathbf{B}_U) is placed along

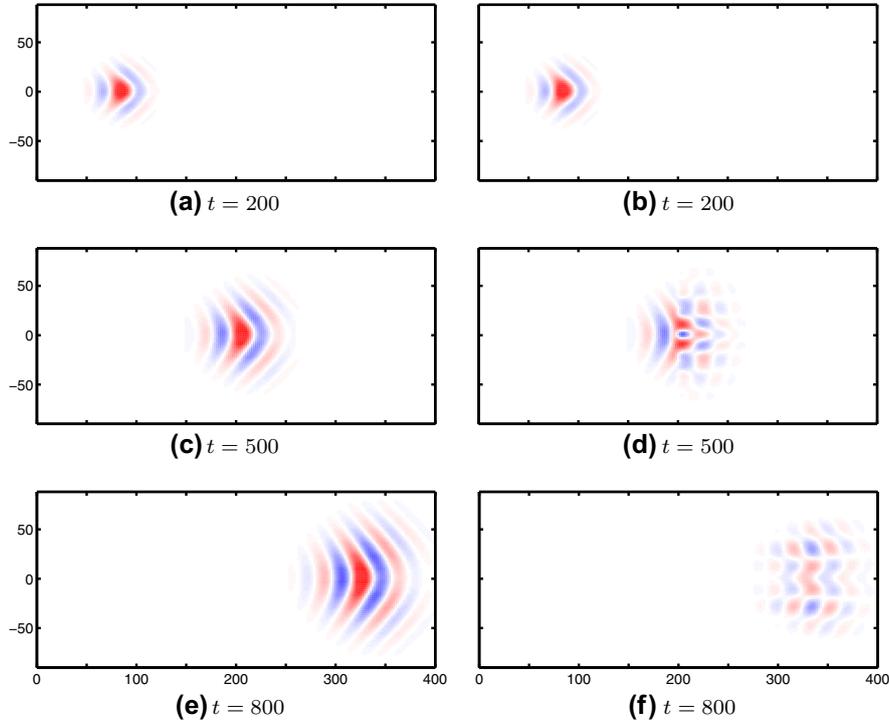


Fig. 5 The propagation of a wavepacket in a flow governed by the modified Kuramoto–Sivashinsky equation (a, c, e) is compared with the controlled case (b, d, f). The non-dimensional parameters in Eqs. 32 and 33 are chosen for mimicking the propagation of the Tollmien–Schlichting waves at $Re = 1000$, with $\alpha_{\max} = 1.68 \times 10^{-1}$, $\beta_{\max} = 2.15 \times 10^{-1}$ and $\omega_{\max} = 2.67 \times 10^{-3}$

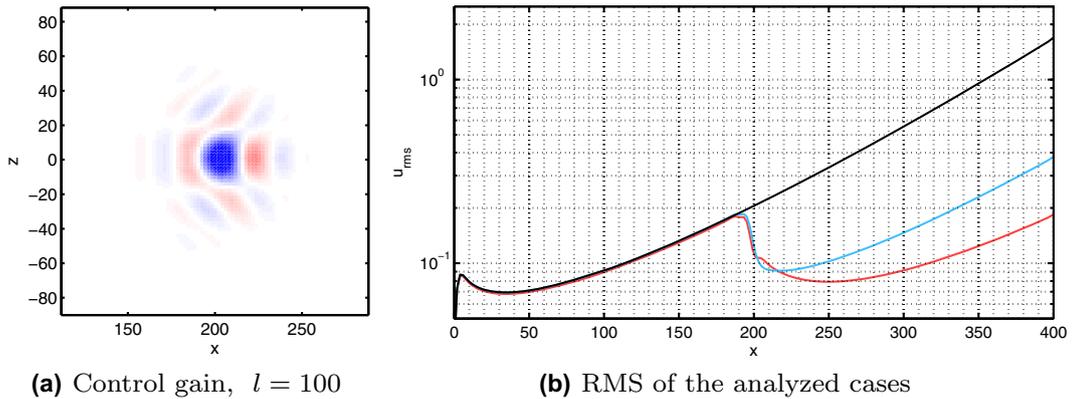


Fig. 6 Left: full-order controller associated with the actuator B_U placed at $x = (200, 0)$ and with $l = 100$. Right: the root mean square (RMS) of the velocity is shown as a function of the streamwise direction x . The value is computed using $N = 10000$ snapshots. The clean configuration (black, solid line) is compared to the controlled cases designed using $l = 100$ (red solid line) and $l = 500$ (blue solid line) (color figure online)

the spanwise direction at $x = 200$, equi-spaced of $\Delta z = 20$. The same setup along the spanwise direction is chosen for the sensors C_z , placed at $x = 300$. All the chosen elements are modeled as Gaussian distributions, Eq. 34.

A centralized controller is designed with $m = 9$ actuators and $p = 9$ sensors. Nine iterations are set, one for each row of the control gain. An example of the results is given in Fig. 6a, where the control gain is placed at $x = (200, 0)$; the controller shows the typical signature of the adjoint solution, as it is bent backward in the plane with respect to the direction of propagation x . Performances of the controllers are shown in Fig. 6b in terms of energy evolution in time; the clean configuration is compared with two cases: $l = 100$ (red solid line), and $l = 500$ (blue solid line). The results are in qualitative agreement with [6].

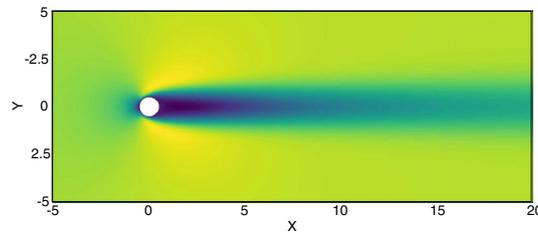


Fig. 7 Streamwise component of the baseflow at $Re = 50$. The cylinder is placed in $(x_0, y_0) = (0, 0)$. The diameter is unitary

6.2 Cylinder flow

The second numerical case is the flow developing behind a two-dimensional cylinder. The literature for this case is rather rich, and we refer to the detailed stability analysis contained in [37,38] (and references therein) for a deeper discussion. For our example, the dynamics of the incompressible flow is governed by the Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \nabla P + \frac{1}{Re_D} \Delta \mathbf{u}, \quad (35)$$

$$0 = \nabla \cdot \mathbf{u}, \quad (36)$$

where \mathbf{u} indicates the velocity field, while P is the pressure. The diameter D is assumed as reference length such that the Reynolds number is defined as $Re_D = UD/\nu$, where ν is the kinematic viscosity. We consider the supercritical case at $Re_D = 50$. The full set of Navier–Stokes equations is solved for the computation of the baseflow \mathbf{U}_{bf} ; since we are considering the supercritical case after the first bifurcation at $Re \approx 47.6$, we apply the selective frequency damping method for the computation of the baseflow, [39]. The residual norm of the final baseflow has been chosen below 10^{-9} .

The Navier–Stokes equations linearized around the baseflow \mathbf{U}_{bf} , and the corresponding adjoint equations, are used for the control design and its verification. For both cases, the equations are formulated first in continuous form and then discretized.

The computational domain extends in the interval $x \in [-20, 50]$ along the streamwise direction, and $y \in [-20, 20]$ along the normal direction, with $(x_0, y_0) = (0, 0)$ being the cylinder position. The Navier–Stokes equations are solved using the Nek5000 code [40], based on the spectral element method [41]. In each cell, the velocity field is defined on Gauss–Legendre–Lobatto points and approximated using Lagrange polynomials of order N , while the pressure is computed on a grid of order $N - 2$ using the same approximation ($\mathbb{P}_N - \mathbb{P}_{N-2}$); the results are obtained with a grid of order $N = 8$. The final mesh is composed by $N_c = 2520$ cells for a total of $N_{gp} = N_c \times N^2$ grid points and approximately $3, 2 \times 10^5$ degrees of freedom. The time integration is based on the BDF3/EXT3 scheme: the convective term is integrated explicitly using a third-order scheme (EXT3), while the diffusion is solved using a backward differentiation (BDF3). Symmetric boundary conditions are imposed along $x = -20$ and $x = 20$, while Neumann conditions are imposed at the outflow. For the inflow, unitary velocity and null perturbation velocity are imposed for the nonlinear simulations and linearized simulations, respectively. The streamwise component of the resulting baseflow is shown in Fig. 7.

6.2.1 Control

The control-unit consists of $m = 2$ actuators \mathbf{B}_u and $p = 2$ sensors \mathbf{C}_z . We consider Gaussian-shaped functions for describing the spatial distribution of the actuators/sensors pairs

$$g(x_0, y_0) = \exp\left(-\frac{(x - x_0)^2}{\sigma^2} - \frac{(y - y_0)^2}{\sigma^2}\right), \quad (37)$$

with $\sigma = 0.5$. The actuators \mathbf{B}_u are located at $(x_0, y_0) = (0.75, \pm 0.75)$, while the sensors \mathbf{C}_z are located further downstream at $(x_0, y_0) = (3.00, \pm 0.75)$. The actuators act only along the streamwise velocity and are approximately located in the region where the structural sensitivity is higher [37]. Sensor measurements are taken only in the streamwise direction. The control penalty of the LQR is $l^2 = 50$. Note that we do not optimize

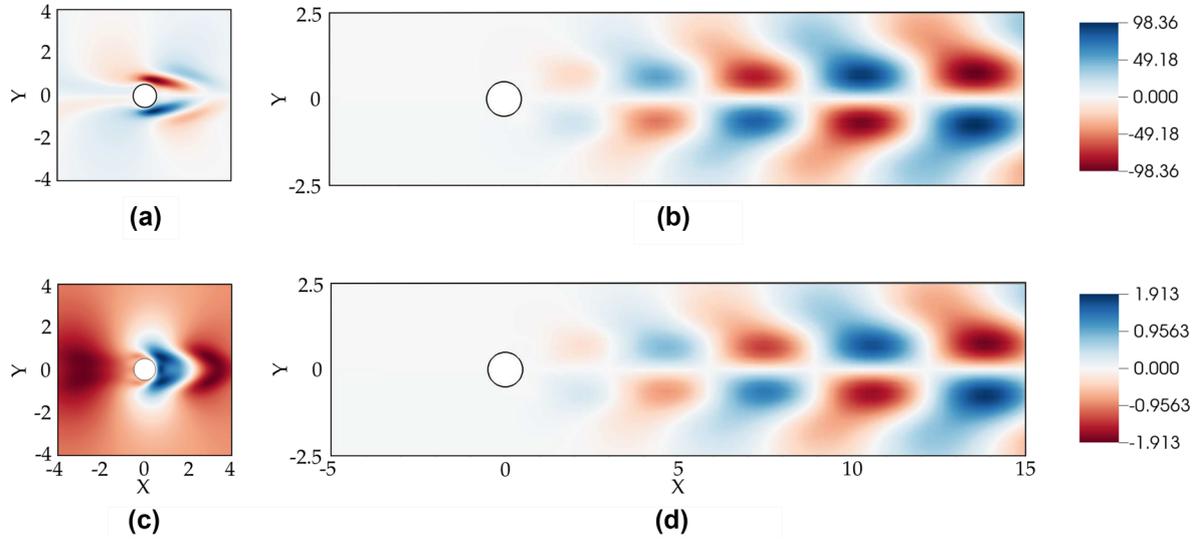


Fig. 8 Left panels: the streamwise (a) and normal (c) components of the control kernel K are shown, for the actuator $B(0.75, 0.75)$. It is possible to note the classical features of the adjoint solution and the localization in the region where the structural sensitivity is higher. Right panels: uncontrolled (b) versus controlled (d) flow at $t = 350$; note that the two cases show a difference of almost two orders of magnitude (on the maximum amplification of the perturbations)

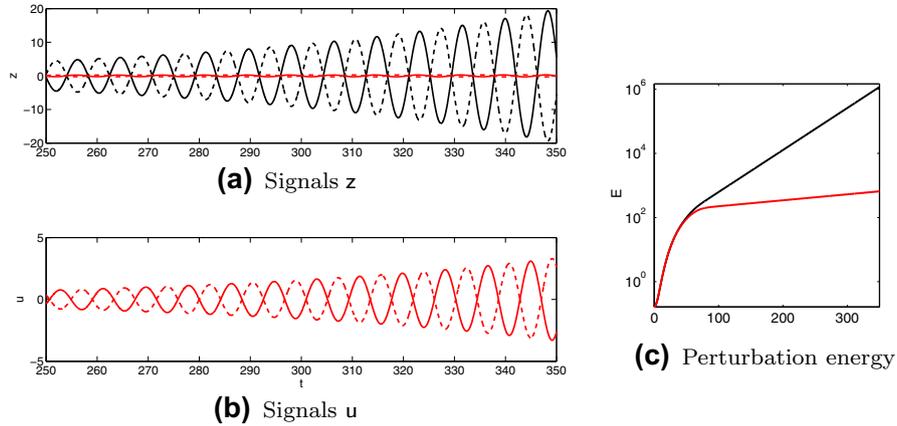


Fig. 9 Quantitative analysis. In all figures, the black lines indicate results of the uncontrolled configuration while the red lines indicate the controlled case. In the panel (a) we show the signals taken at $C_z(3, 0.75)$ —solid line—and $C_z(3, -0.75)$ —dashed line—compared with the corresponding signals recorded when the control is active. In the right panel (c), the energy perturbation for the two cases is shown. In the panel (b), the control signals feeding $B_u(0.75, 0.75)$ —solid line—and $B_u(0.75, -0.75)$ —dashed line—are shown (color figure online)

the choice of l or the location/distribution of the actuator/sensor pairs; the main goal is to show the feasibility of the approach in designing a multiple inputs, full-order controller also for systems where the Riccati solution is not available.

The ADA iteration is performed using a fixed step $\beta = 0.01$ for the update in Eq. 6. The loop terminates when the norm of the gradient $\|\partial\mathcal{J}/\partial u\|$ is below a tolerance of $O(10^{-8})$. For this case, $N_i = 19$ iterations were sufficient for reaching this threshold. In Fig. 8, the streamwise component (a) and the normal component (c) of the controller associated with the actuator $B_u(0.75, 0.75)$ are shown; the resulting solution shows the typical features of the adjoint structures. Moreover, the spatial distribution of the control gain is localized in both the regions $y > 0$ and $y < 0$. This is due to the choice of a centralized controller; indeed, the adjoint solution is forced at the locations of both the sensors and the coupling among the actuators is not discarded. Also, note that the control kernel approximately self-localizes in the region where the sensitivity is higher.

The control performances are analyzed by comparing the evolution of the linear perturbations at $Re_D = 50$ with the simulation where the controller is turned on at $t = 20$. In Fig. 8b, d, we show two snapshots taken

at $t = 350$, for the clean configuration and the controlled case, respectively. As it is possible to verify, the vortex street is damped by two orders of magnitude. In Fig. 9, the comparison is performed quantitatively by considering the signals recorded in C_z and the perturbation energy, computed as the weighted inner-product of the perturbation velocities. In the figures, the controlled case is always indicated with red lines, while the uncontrolled case is indicated with black lines. When the controller is turned on, the perturbation energy (Fig. 9c) and the signals z (Fig. 9a) quickly drop, while the clean configuration exploits the exponential energy growth. Finally, in Fig. 9b the control signals are shown.

7 Conclusions and discussion

In this work, we extended the *Adjoint of the direct-adjoint* (ADA) algorithm for the computation of full-order controllers in large systems. In particular, we address the cases with multiple inputs/outputs, by highlighting how decentralized, centralized and coupled controllers can be computed. The analysis is performed both analytically and numerically.

First, we demonstrate the equivalence between the solution of the Riccati equation and the ADA solution using a scalar problem. Similarly to matrix-free techniques based on the time-steppers, we observe that the iterative scheme approximates the action of the Riccati solution. The result extends also to large systems. These observations apply to the dual problem—i.e., the estimation problem—and to robust controllers belonging to the \mathcal{H}_∞ framework.

In the second part of the manuscript, we consider numerical examples. A toy-problem mimicking a distributed system is analyzed for assessing the convergence performance of ADA against an analogous algorithm by Mårtensson et al. [35], indicated as MR. The results show a remarkable scalability of ADA, with respect to the number of actuators. Finally, examples from fluid mechanics applications are shown using the linearized Kuramoto–Sivashinsky, representative of a subcritical flow, and the flow past a two-dimensional cylinder at $Re = 50$. The feasibility of the approach is shown by designing multiple inputs LQR controllers for both cases.

As final discussion, we suggest some future directions based on this work and stress some of the shortcomings of this approach. Starting from the limitations of the ADA approach, it is important to note that this application is restricted to plants in numerical settings where a full model is available. In general, an accurate model of an experimental settings is not available unless using a reduced-order model approach. Also, when it is possible to apply controllers based on numerical models to experiments, the reduction step is still necessary; in this perspective, one of the main problems in control design is the robustness to uncertainties of the model or the robustness of the closed-loop system with respect to off-design operative conditions (see for instance [42]). Using ADA, we tackle the optimization aspect of the closed-loop control problem, but not directly the robustness issues, for which we refer to the recent review by [5]. Among the advantages of the algorithm, we can certainly mention: i) the possibility of assessing preliminarily the optimal performance of a controlled plant, at the design point; ii) the independence of the final solution with respect to the initial conditions, implying that it is not required by any knowledge of the disturbances active in the flow for the control problem; iii) the easiness of implementation of the scheme, largely based on a standard direct-adjoint loop. Within this perimeter, future work will be devoted to the optimization of the sensor/actuator pairs coupling, their optimal placement, and the numerical solution of the full-order \mathcal{H}_∞ problem. Indeed, the sensors/actuators pairing and their placement in our application is predetermined and not optimized, while convergence issues can arise when a *min-max* optimization is applied for the robust optimal control. For both cases, modern techniques of optimization might be a successful ingredient; an example is given by the sparsity promoting algorithms (see [43]), potentially well-fitted for the pairing problem.

Acknowledgements Anders Rantzer and his collaborators (Lund University—SE) are acknowledged for bringing up their work on optimal control synthesis in large systems. Nicolò Fabbiane (DAAA-ONERA, Meudon—FR) and Michele Alessandro Bucci (DynFluid-ENSAM, Paris—FR) are acknowledged for providing the Kuramoto–Sivashinsky scripts and for the help in setting the test-case in Nek5000, respectively. This work was developed mostly when the first author served as postdoctoral researcher at the *Laboratoire d'Hydrodynamique de l'Ecole polytechnique* (LadHyX, Palaiseau - FR); the support by the Agence Nationale de la Recherche (*CoolJazz* project, grant number ANR-12-BS09-0024) and by the DGA (grant number 2015.60.0004) are acknowledged.

A Method of the unit solutions

We demonstrated in Sect. 3.1 the equivalence between the ADA method and the Riccati equation using the analytical solution based on the method of the *unit solutions* ([7, 29]). The method allows us to calculate the solution of the problem starting from two initial-value problems, reported in the following.

Initial-value problem I We consider as the initial conditions $v_1(t_0) = v_0$ and $\lambda_1(t_0) = 0$, resulting in the solution

$$\begin{bmatrix} v_1(t) \\ \lambda_1(t) \end{bmatrix} = \frac{v_0}{2\alpha} \begin{bmatrix} \alpha - a \\ w \end{bmatrix} e^{-\alpha t} + \frac{v_0}{2\alpha} \begin{bmatrix} \alpha + a \\ -w \end{bmatrix} e^{\alpha t}, \quad (38)$$

where $\alpha = \sqrt{a^2 + w\psi}$ corresponds to one of the eigenvalues of the Hamiltonian problem in Eq. 13.

Initial-value problem II We consider as the initial conditions $v_2(t_0) = 0$ and $\lambda_2(t_0) = 1$. In a more general case, with N degrees of freedom, a number of independent initial conditions is considered such that $\lambda_{i+1} = e_i$ with $i = 1 \dots N$. Here, we consider only one solution

$$\begin{bmatrix} v_2(t) \\ \lambda_2(t) \end{bmatrix} = \frac{1}{2\alpha} \begin{bmatrix} \psi \\ \alpha + a \end{bmatrix} e^{-\alpha t} + \frac{1}{2\alpha} \begin{bmatrix} -\psi \\ \alpha - a \end{bmatrix} e^{\alpha t}. \quad (39)$$

A general solution of the problem is obtained as a linear combination of the initial-value problems

$$\begin{aligned} \begin{bmatrix} v(t) \\ \lambda(t) \end{bmatrix} &= \begin{bmatrix} v_1(t) \\ \lambda_1(t) \end{bmatrix} + c \begin{bmatrix} v_2(t) \\ \lambda_2(t) \end{bmatrix} \\ &= \frac{1}{2\alpha} \begin{bmatrix} (\alpha - a)v_0 + c\psi \\ wv_0 + c(\alpha + a) \end{bmatrix} e^{-\alpha t} + \frac{1}{2\alpha} \begin{bmatrix} (\alpha + a)v_0 - c\psi \\ -wv_0 + c(\alpha - a) \end{bmatrix} e^{\alpha t}, \end{aligned}$$

where the constant c is unknown. Recombining the solutions we get

$$v(t) = \frac{(\alpha - a)v_0 + c\psi}{2\alpha} e^{-\alpha t} + \frac{(\alpha + a)v_0 - c\psi}{2\alpha} e^{\alpha t}, \quad (40)$$

$$\lambda(t) = \frac{wv_0 + c(\alpha + a)}{2\alpha} e^{-\alpha t} + \frac{-wv_0 + c(\alpha - a)}{2\alpha} e^{\alpha t}. \quad (41)$$

Finally, we impose the 4 terminal conditions at $t = 0$ and $t = T$ for the initial-value problem. By introducing the hyperbolic functions, the direct solution v and the adjoint λ at $t = 0$ and $t = T$ are written as

$$v(0) = v_0 \quad (42)$$

$$\lambda(0) = c \quad (43)$$

$$v(T) = \alpha v_0 \cosh(\alpha T) + (av_0 - c\psi) \sinh(\alpha T) \quad (44)$$

$$\lambda(T) = -(wv_0 + ca) \sinh(\alpha T) + c\alpha \cosh(\alpha T). \quad (45)$$

These are the solutions of the optimal control problem as stated in Sect. 2. For the ADA case, we introduce $v(0) = v_0 = -b_i/r_i$ and $\lambda(T) = 0$. Thus, the constant c is

$$c = \frac{wv_0}{\alpha - a \tanh(\alpha T)} \tanh(\alpha T). \quad (46)$$

The demonstration of the equivalence finalizes by observing that the control kernel $k_i = \lambda(0) = c$. The exact solution of the optimal control problem is obtained for $T \rightarrow \infty$.

B Derivation of ADA algorithm using integration by parts

In this section, we briefly summarize the original derivation of the ADA algorithm, as first proposed by [1]. We start from the direct-adjoint system

$$\frac{d\mathbf{v}}{dt} = \mathbf{A}\mathbf{v} - \Psi\boldsymbol{\lambda}, \quad \text{with } \mathbf{v}(0) = \mathbf{v}_0, \quad (47a)$$

$$\frac{d\boldsymbol{\lambda}}{dt} = -\mathbf{A}^H\boldsymbol{\lambda} - \mathbf{W}\mathbf{v}, \quad \text{with } \boldsymbol{\lambda}(T) = 0, \quad (47b)$$

where $\Psi = \mathbf{B}_u\mathbf{R}^{-1}\mathbf{B}_u^H$ and the optimality condition has already been included. By introducing two variables, namely the state vectors $\mathbf{a}(t)$ and $\mathbf{b}(t)$, we can recombine the system as

$$\int_0^T \mathbf{a}^H \left(\frac{d\mathbf{v}}{dt} - \mathbf{A}\mathbf{v} - \Psi\boldsymbol{\lambda} \right) dt + \int_0^T \mathbf{b}^H \left(\frac{d\boldsymbol{\lambda}}{dt} + \mathbf{A}^H\boldsymbol{\lambda} + \mathbf{W}\mathbf{v} \right) dt.$$

Integration by parts allows rewriting the relation as

$$\int_0^T \mathbf{v}^H \left(-\frac{d\mathbf{a}}{dt} - \mathbf{A}^H\mathbf{a} + \mathbf{W}\mathbf{b} \right) dt + \int_0^T \boldsymbol{\lambda}^H \left(-\frac{d\mathbf{b}}{dt} + \mathbf{A}\mathbf{b} - \Psi\mathbf{a} \right) dt \\ + (\mathbf{a}^H\mathbf{v} + \mathbf{b}^H\boldsymbol{\lambda})_T - (\mathbf{a}^H\mathbf{v} + \mathbf{b}^H\boldsymbol{\lambda})_0$$

Gathering the state equations for $\mathbf{a}(t)$ and $\mathbf{b}(t)$, and changing the variables $\mathbf{a} \rightarrow \tilde{\boldsymbol{\lambda}}$, $\mathbf{b} \rightarrow -\tilde{\mathbf{v}}$, allow us to write the following system

$$\frac{d\tilde{\mathbf{v}}}{dt} = \mathbf{A}\tilde{\mathbf{v}} + \Psi\tilde{\boldsymbol{\lambda}}, \quad \tilde{\mathbf{v}}(0) = \tilde{\mathbf{v}}_0, \quad (48a)$$

$$-\frac{d\tilde{\boldsymbol{\lambda}}}{dt} = \mathbf{A}^H\tilde{\boldsymbol{\lambda}} + \mathbf{W}\tilde{\mathbf{v}}, \quad \tilde{\boldsymbol{\lambda}}(T) = 0, \quad (48b)$$

$$\tilde{\boldsymbol{\lambda}}(T)^H\mathbf{v}(T) - \tilde{\mathbf{v}}(T)^H\boldsymbol{\lambda}(T) = \tilde{\boldsymbol{\lambda}}(0)^H\mathbf{v}(0) - \tilde{\mathbf{v}}(0)^H\boldsymbol{\lambda}(0). \quad (48c)$$

Equation 48 corresponds to the ones of the original optimal control problem 47. The last relation 48(c) is exactly the condition recovered by symplectic product in Eq. (11); thus, the observations done in Eqs. (11–12) are valid also for this case.

C Full-order controllers using stochastic gradients

In this appendix, the algorithm discussed by Mårtensson and Rantzer [35] is derived following the Lagrangian approach. For sake of conciseness, we will indicate the algorithm with the acronym MR. The state equation is rewritten as

$$\frac{d\mathbf{v}}{dt} = (\mathbf{A} + \mathbf{B}_u\mathbf{K})\mathbf{v} \quad \text{with } \mathbf{v}(0) = \mathbf{v}_0, \quad (49)$$

The problem is rewritten now assuming the control kernel \mathbf{K} to be the unknown of the problem. The cost function is now

$$\mathcal{J} = \frac{1}{2} \int_0^T \left(\mathbf{v}^H \mathbf{W}\mathbf{v} + \mathbf{v}^H \mathbf{K}^H \mathbf{R}\mathbf{K}\mathbf{v} \right) dt = \frac{1}{2} \int_0^T \mathbf{v}^H \left(\mathbf{W} + \mathbf{K}^H \mathbf{R}\mathbf{K} \right) \mathbf{v} dt, \quad (50)$$

and the resulting augmented Lagrangian is

$$\tilde{\mathcal{J}} = \mathcal{J} - \int_0^T \boldsymbol{\lambda}^H \left(\frac{d\mathbf{v}}{dt} - \mathbf{A}\mathbf{v} - \mathbf{B}_u\mathbf{K}\mathbf{v} \right) dt.$$

The resulting system is obtained by considering the gradients with respect to λ , v and K

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \lambda} = \frac{dv}{dt} - Av - B_u K v, \quad (51a)$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial v} = \frac{d\lambda}{dt} + A^H \lambda + K^H B_u^H \lambda + (W + K^H R K) v, \quad (51b)$$

$$\frac{\partial \tilde{\mathcal{J}}}{\partial K} = \int_0^T (R K v + B_u^H \lambda) v^H dt. \quad (51c)$$

Zeroing the gradients, the system can be arranged in matrix form as

$$\begin{pmatrix} \dot{v} \\ \dot{\lambda} \end{pmatrix} = \begin{bmatrix} A + B_u K & \mathbf{0} \\ -(W + K^H R K) - (A + B_u K)^H \end{bmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix}.$$

The solution is updated by using 51c as

$$K^{i+1} = K^i - \beta \nabla_K \tilde{\mathcal{J}}. \quad (52)$$

The iteration is usually initialized with a guess for K , unless the system under consideration is asymptotically stable; in the latter case, the controller is designed for modifying the transient dynamics and the starting guess is simply a null vector. In this formulation, the problem depends on the initial conditions. The solution is not guaranteed to be the one corresponding to the global minimum for a given initial condition. This drawback is circumvented by using ad-hoc strategies as the stochastic gradient approach described in [35]. Moreover, with respect to the standard algorithm, the MR algorithm does not require the solution of Lyapunov equations, replaced with an approximation based on the adjoint and direct solutions.

C.1 Relation with the ADA algorithm: generalizing the sensitivity approach

The symplectic product adopted in ADA can be introduced also for the system in Eq. 51. It is possible to show that the relation

$$\tilde{\lambda}(t)^H v(t) = \tilde{v}(t)^H \lambda(t) \quad \forall t, \quad (53)$$

is fulfilled also in this case. Considering again the gradient,

$$\nabla_K \tilde{\mathcal{J}} = \int_0^T (R K v + B_u^H \lambda) v^H dt, \quad (54)$$

it is null $\forall v$ if

$$R K v(t) = -B_u^H \lambda(t). \quad (55)$$

By rescaling the relation via R^{-1} , we recover the equivalence between the product and the optimality condition at $t = 0$

$$\tilde{\lambda}(0)^H v_0 = \tilde{v}(0)^H \lambda_0, \quad (56a)$$

$$K v_0 = -R^{-1} B_u^H \lambda_0. \quad (56b)$$

Thus, also for the basic equation of the MR algorithm, introducing as initial conditions of the system $\tilde{q} = -R_{i,i}^{-1} B_{u,i}$, the iteration enables to compute the i -th kernel K as solution of the adjoint equation at the final time T . All the observations done for the multiple inputs, coupled version of the ADA algorithm are valid also for this algorithm. Interestingly, this algorithm produces the control kernels in two different ways: i) as adjoint solution; ii) as result of the gradient-based iteration

$$K^{n+1} = K^n - \beta \nabla_K \mathcal{J}. \quad (57)$$

The sensitivity with respect to the initial conditions applied to the optimal control problem leads to the same results with different choice of gradients. The gradient with respect to \mathbf{K} can be rewritten, using the chain-rule, as

$$\frac{\partial \tilde{\mathcal{J}}}{\partial \mathbf{K}} = \frac{\partial \tilde{\mathcal{J}}}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{K}} = \left(\mathbf{R} \mathbf{K} \mathbf{v} + \mathbf{B}_u^H \boldsymbol{\lambda} \right) \mathbf{v}^H. \quad (58)$$

The underlying system of equations is equivalent in the two cases.

References

1. Bewley, T., Luchini, P., Pralits, J.O.: Methods for solution of large optimal control problems that bypass open-loop model reduction. *Meccanica* **51**(12), 2997–3014 (2016). <https://doi.org/10.1007/s11012-016-0547-3>
2. Abergel, F., Temam, R.: On some control problems in fluid mechanics. *Theor. Comput. Fluid Dyn.* **1**, 303–325 (1990). <https://doi.org/10.1007/BF00271794>
3. Kim, J., Bewley, T.R.: A linear systems approach to flow control. *Ann. Rev. Fluid Mech.* **39**, 39–383 (2007). <https://doi.org/10.1146/annurev.fluid.39.050905.110153>
4. Fabbiane, N., Semeraro, O., Bagheri, S., Henningson, D.S.: Adaptive and model-based control theory applied to convectively unstable flows. *Appl. Mech. Rev.* **66**(6), 060801 (2014). <https://doi.org/10.1115/1.4027483>
5. Sipp, D., Schmid, P.J.: Linear closed-loop control of fluid instabilities and noise-induced perturbations: a review of approaches and tools. *Appl. Mech. Rev.* **68**(2), 020801 (2016). <https://doi.org/10.1115/1.4033345>
6. Semeraro, O., Bagheri, S., Brandt, L., Henningson, D.S.: Transition delay in a boundary layer flow using active control. *J. Fluid Mech.* **731**, 288–311 (2013). <https://doi.org/10.1017/jfm.2013.299>
7. Lewis, F.L., Vrabie, D., Syrmos, V.L.: *Optimal Control*. Wiley, Hoboken (2012)
8. Benner, P., Li, J.R., Penzl, T.: Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems. *Numer. Linear Algebra Appl.* **15**, 755–777 (2008). <https://doi.org/10.1002/nla.622>
9. Benner, P.: Solving large-scale control problems. *IEEE Control Syst. Mag.* **24**(1), 44–59 (2004). <https://doi.org/10.1109/MCS.2004.1272745>
10. Banks, H.T., Ito, K.: A numerical algorithm for optimal feedback gains in high dimensional linear quadratic regulator problems. *SIAM J. Control Optim.* **29**(3), 499–515 (1991). <https://doi.org/10.1137/0329029>
11. Akhtar, I., Borggaard, J., Burns, J.A., Imtiaz, H., Zietsman, L.: Using functional gains for effective sensor location in flow control: a reduced-order modelling approach. *J. Fluid Mech.* **781**, 622–656 (2015). <https://doi.org/10.1017/jfm.2015.509>
12. Högberg, M., Bewley, T.R., Henningson, D.S.: Linear feedback control and estimation of transition in plane channel flow. *J. Fluid Mech.* **481**, 149–175 (2003). <https://doi.org/10.1017/S0022112003003823>
13. Bamieh, B., Paganini, F., Dahleh, M.A.: Distributed control of spatially invariant systems. *IEEE Trans. Autom. Control* **47**(7), 1091–1107 (2002). <https://doi.org/10.1109/TAC.2002.800646>
14. Chevalier, M., Hoepffner, J., Åkervik, E., Henningson, D.S.: Linear feedback control and estimation applied to instabilities in spatially developing boundary layers. *J. Fluid Mech.* **588**, 163–187 (2007). <https://doi.org/10.1017/S0022112007007392>
15. Monokrousos, A., Brandt, L., Schlatter, P., Henningson, D.S.: Dns and les of estimation and control of transition in boundary layers subject to free-stream turbulence. *Int. J. Heat Fluid Flow* **29**(3), 841–855 (2008). <https://doi.org/10.1016/j.ijheatfluidflow.2008.03.009>
16. Carini, M., Pralits, J.O., Luchini, P.: Feedback control of vortex shedding using a full-order optimal compensator. *J. Fluid Struct.* **53**, 15–25 (2015). <https://doi.org/10.1016/j.jfluidstructs.2014.11.011>
17. Carini, M., Airiau, C., Debien, A., Léon, O., Pralits, J.O.: Global stability and control of the confined turbulent flow past a thick flat plate. *Phys. Fluids* **29**(2), 024102 (2017). <https://doi.org/10.1063/1.4974069>
18. Luchini, P., Bottaro, A.: Adjoint equations in stability analysis. *Ann. Rev. Fluid* **46**, 493–517 (2014). <https://doi.org/10.1146/annurev-fluid-010313-141253>
19. Bewley, T.R., Moin, P., Temam, R.: DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms. *J. Fluid Mech.* **447**(2), 179–225 (2001). <https://doi.org/10.1017/S0022112001005821>
20. Cherubini, S., Robinet, J.-C., De Palma, P.: Nonlinear control of unsteady finite-amplitude perturbations in the Blasius boundary-layer flow. *J. Fluid Mech.* **737**, 440–465 (2013). <https://doi.org/10.1017/jfm.2013.576>
21. Kim, J., Bodony, D.J., Freund, J.B.: Adjoint-based control of loud events in a turbulent jet. *J. Fluid Mech.* **741**, 28–59 (2014). <https://doi.org/10.1017/jfm.2013.654>
22. Semeraro, O., Pralits, J.O., Rowley, C., Henningson, D.S.: Riccati-less approach for optimal control and estimation: an application to two-dimensional boundary layers. *J. Fluid Mech.* **731**, 394–417 (2013). <https://doi.org/10.1017/jfm.2013.352>
23. Zhou, K., Doyle, J.C., Glover, K.: *Robust and Optimal Control*. Prentice Hall, Upper Saddle River (2002)
24. Mårtensson, K.: *Gradient Methods for Large-Scale and Distributed Linear Quadratic Control*. Ph.D. thesis, Lund University (2012)
25. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd edn. Cambridge University Press, Cambridge (2007)
26. Laub, A.: A schur method for solving algebraic Riccati equations. *IEEE Trans. Autom. Control* **24**(6), 913–921 (1979)
27. Glad, T., Ljung, L.: *Control Theory*. CRC Press, Boca Raton (2000)
28. Fabbiane, N., Bagheri, S., Henningson, D.S.: Energy efficiency and performance limitations of linear adaptive control for transition delay. *J. Fluid Mech.* **810**, 60–81 (2017). <https://doi.org/10.1017/jfm.2016.707>

29. Bryson, A.E.: Applied Optimal Control: Optimization, Estimation and Control. CRC Press, Boca Raton (1975)
30. Tuckerman, L.S., Barkley, D.: Bifurcation analysis for timesteppers. In: Numerical Methods for Bifurcation Problems and Large-Scale Dynamical Systems, pp. 453–466. Springer (2000). https://doi.org/10.1007/978-1-4612-1208-9_20
31. Bagheri, S., Brandt, L., Henningson, D.S.: Input–output analysis, model reduction and control of the flat-plate boundary layer. *J. Fluid Mech.* **620**, 263–298 (2009). <https://doi.org/10.1017/S0022112008004394>
32. Bagheri, S., Høpfner, J., Schmid, P.J., Henningson, D.S.: Input-output analysis and control design applied to a linear model of spatially developing flows. *Appl. Mech. Rev.* (2009). <https://doi.org/10.1115/1.3077635>
33. Gumussoy, S., Henrion, D., Millstone, M., Overton, M.L.: Multiobjective robust control with HIFOO 2.0. *IFAC Proc.* **42**(6), 144–149 (2009). <https://doi.org/10.3182/20090616-3-IL-2002.00025>
34. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3**(1), 1–122 (2011). <https://doi.org/10.1561/22000000016>
35. Mårtensson, K., Rantzer, A.: A scalable method for continuous-time distributed control synthesis. In: American Control Conference (ACC), pp. 6308–6313. IEEE (2012). <https://doi.org/10.1109/ACC.2012.6314762>
36. Sundin, J.: Controlling the Laminar-to-Turbulent Transition in a Fluid Flow. Tech. rep, KTH (2015)
37. Giannetti, F., Luchini, P.: Structural sensitivity of the first instability of the cylinder wake. *J. Fluid Mech.* **581**, 167–197 (2007). <https://doi.org/10.1017/S0022112007005654>
38. Marquet, O., Sipp, D., Jacquin, L.: Sensitivity analysis and passive control of cylinder flow. *J. Fluid Mech.* **615**, 221–252 (2008). <https://doi.org/10.1017/S0022112008003662>
39. Åkervik, E., Brandt, L., Henningson, D.S., Høpfner, J., Marxen, O., Schlatter, P.: Steady solutions of the Navier–Stokes equations by selective frequency damping. *Phys. Fluids* **18**(6), 068102 (2006). <https://doi.org/10.1063/1.2211705>
40. Fischer, P.F., Lottes, J.W., Kerkemeier, S.G.: nek5000 web page, Web page: <http://nek5000.mcs.anl.gov>
41. Patera, A.T.: A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J. Comput. Phys.* **54**(3), 468–488 (1984). [https://doi.org/10.1016/0021-9991\(84\)90128-1](https://doi.org/10.1016/0021-9991(84)90128-1)
42. Fabbiane, N., Simon, B., Fischer, F., Grundmann, S., Bagheri, S., Henningson, D.S.: On the role of adaptivity for robust laminar flow control. *J. Fluid Mech.* (2015). <https://doi.org/10.1017/jfm.2015.45>
43. Lin, F., Fardad, M., Jovanović, M.R.: Design of optimal sparse feedback gains via the alternating direction method of multipliers. *IEEE Trans. Autom. Control* **58**(9), 2426–2431 (2013). <https://doi.org/10.1109/TAC.2013.2257618>