Continuous-time balanced truncation for time-periodic fluid flows using frequential Gramians

Alberto Padovan *, Clarence W. Rowley

Mechanical and Aerospace Engineering Department, Princeton University, Olden St., 08544, Princeton, NJ, USA

ARTICLE INFO

ABSTRACT

Reduced-order models for flows that exhibit time-periodic behavior (e.g., flows in turbomachinery and wake flows) are critical for several tasks, including active control and optimization. One well-known procedure to obtain the desired reduced-order model in the proximity of a periodic solution of the governing equations is continuous-time balanced truncation. Within this framework, the periodic reachability and observability Gramians are usually estimated numerically via quadrature using the forward and adjoint post-transient response to impulses. However, this procedure can be computationally expensive, especially in the presence of slowly-decaying transients. Moreover, it can only be performed if the periodic orbit is stable in the sense of Floquet. In order to address these issues, we use the frequency-domain representation of the Gramians, which we henceforth refer to as *frequential Gramians*. First, these frequential Gramians are well-defined for both stable and unstable dynamics. In particular, we show that when the underlying system is unstable, these Gramians satisfy a pair of allied differential Lyapunov equations. Second, they can be estimated numerically by solving algebraic systems of equations that lend themselves to heavy computational parallelism and that deliver the desired post-transient response without having to follow physical transients. The computational gains that we can achieve by using the frequency domain are demonstrated on a simple three-dimensional toy model that exhibits time-periodic dynamics. We then demonstrate this method on a periodically-forced axisymmetric jet at Reynolds numbers $Re = 1250$ and $Re = 1500$. At the lower Reynolds number, the flow strongly amplifies subharmonic perturbations and exhibits vortex pairing about a Floquet-stable T-periodic solution. At the higher Reynolds number, the underlying T-periodic orbit is unstable and the flow naturally settles onto a $2T$-periodic limit cycle characterized by pairing vortices. At both Reynolds numbers, we compute a reduced-order model and we use it to design a feedback controller and a state estimator capable of suppressing vortex pairing.

1. Introduction

Physical processes are often governed by partial differential equations, which, upon spatial discretization, lead to high-dimensional systems of ordinary differential equations. Although recent advances in computational resources have allowed us to simulate these systems quite efficiently, tasks such as controller design and optimization can seldom be performed in the original high-dimensional space. It therefore becomes necessary to develop low-order models that capture the salient features of the underly-

ing dynamics. In this paper, we focus on systems that exhibit time-periodic behavior, and we seek a low-order representation of the dynamics in the proximity of a time-periodic solution of the governing equations.

While there are many existing methods for model reduction of both linear and nonlinear systems, here we provide an overview of the linear techniques based on "balancing". In the simplest of cases, i.e., the balancing of a linear time-invariant system, one seeks a reduced-order model by first identifying a change of coordinates where two matrices known as reachability and observability Gramians are equal and diagonal. This method, initially introduced by Moore [23], has become increasingly popular because of well-known a-priori error bounds (see, e.g., Dullerud and Paganini [7]) and because of its relatively low computational cost. More recently, the balanced proper orthogonal decomposition (BPOD) framework introduced by Rowley [27] led to a further reduction of the computational cost in systems with a large number of outputs, and the method has since become a benchmark for model reduction of linear systems as well as of nonlinear systems that evolve near a steady state. Although balanced truncation and BPOD were originally conceived for stable systems, they have also been applied to unstable systems upon slight modifications. For instance, Ahuja and Rowley [1] proposed splitting the stable and unstable eigenspaces, and balancing the stable dynamics, while treating the unstable eigenspace exactly. Alternatively, Dergham et al. [6] obtained balanced reduced-order models of an open cavity flow with an underlying unstable steady state by leveraging the frequency-domain representation of the Gramians, which is well-defined for both stable and unstable systems (see, e.g., Godunov [11]). Finally, Flinois et al. [9] showed that the original algorithm developed for stable systems could be used to balance unstable systems without the a-priori splitting of the stable and unstable eigenspaces.

Balanced truncation has also been used for discrete-time periodic systems (see, for instance, Longhi and Orlando [17], Varga [31] and Farhood et al. [8]). More recently, a procedure similar to BPOD was developed by Ma et al. [20] by lifting the discrete-time periodic system into a higher-dimensional linear time-invariant system. That formulation was then applied in Ma [19] for controller design to stabilize an unstable periodic orbit in the wake of a flat plate. The balancing of linear time-varying systems in their continuous-time formulation is discussed in Sandberg and Rantzer [28], and error bounds for monotonically-balanced and non-monotonically-balanced systems are presented. Continuous-time balancing was also performed in Lang et al. [16], where the authors presented an implicit time integration method to solve the differential Lyapunov equations that govern the dynamics of the (time-varying) reachability and observability Gramians. Despite the fact that balancing for periodic systems is well-understood, to the best of our knowledge it is rarely used in practice in very high-dimensional systems such as two-dimensional or three-dimensional fluid flows. In fact, the (discrete-time) application in Ma [19] is the only one we are aware of. This is most likely due to the fact that computing the Gramians for a time-varying system can be expensive. In particular, this requires computing the post-transient response to forward and adjoint impulses, and the computational cost can grow significantly if the underlying dynamics exhibit slowly-decaying transients. Furthermore, this procedure to estimate the Gramians can only be performed on systems that are stable in the sense of Floquet, unless the stable and unstable Floquet eigenspaces are treated separately (as in Ma et al. [20]) at additional computational cost. Here, we propose to address these problems using the frequency-domain representation of the Gramians. We henceforth refer to these Gramians as *frequential Gramians*.

As in time-invariant systems, if the underlying system is stable, the frequential Gramians agree with the time-domain representation of the Gramians. Unlike their time-domain counterparts, the frequential Gramians are also well-defined if the dynamics are unstable, and we show that they satisfy a pair of allied differential Lyapunov equations. Consequently, we can use these Gramians to obtain balanced low-order models of unstable systems, and this is particularly important if we wish to design reduced-order stabilizing controllers. We shall also see that while the frequential Gramians for time-invariant systems are defined in terms of the resolvent operator associated with the underlying system, the frequential Gramians for time-periodic systems are defined in terms of the harmonic resolvent operator [24,25,33].

From a computational standpoint, the use of frequential Gramians can lead to computational savings. Specifically, estimating the Gramians no longer requires performing impulse responses in the time domain, but it simply amounts to solving algebraic systems of equations that lend themselves to heavy computational parallelism and that deliver the desired post-transient solution without having to follow the physical transients. As discussed in section 4, additional savings can be obtained by leveraging some of the symmetries of the harmonic resolvent operator, which, as previously mentioned, is used to define the frequential Gramians. More thorough computational considerations are presented in section 5, where we also present performance results using a three-dimensional toy model that exhibits time-periodic dynamics.

We use this framework to compute reduced-order models for a periodically-forced axisymmetric incompressible jet at Reynolds numbers $Re = 1250$ and $Re = 1500$. At $Re = 1250$, the flow exhibits a Floquet-stable periodic orbit of period T, characterized by an unpaired vortex street. However, as discussed in Shaabani-Ardali et al. [30] and Padovan and Rowley [25], this configuration is extremely sensitive to period-doubling perturbations, so that any small-amplitude perturbation will cause neighboring vortices to merge and pair. Here, we compute a reduced-order model of the dynamics in the proximity of the periodic orbit, and we then design a feedback controller and an observer that successfully suppress vortex pairing in the presence of disturbances. At $Re = 1500$ the underlying T-periodic orbit is linearly unstable, so any small perturbation will grow and eventually settle onto a $2T$-periodic limit cycle characterized by pairing vortices. As before, we compute a reduced-order model and we design a controller and an observer to restabilize the T-periodic orbit and suppress vortex pairing.

Although, to the best of our knowledge, this is the first time that frequential Gramians are used in the balancing of time-periodic systems, it is important to mention related work that leverages the frequency-domain representation of time-periodic systems. For instance, Jovanović and Fardad [14] proposed solving a sequence of simplified Sylvester and Lyapunov equations to approximate the $\mathcal{H}_2$ norm of linear periodically time-varying system, where the periodic component is small. Their approach was then implemented on a pressure-driven channel subject to streamwise oscillations of the bottom wall. The same approach was used in Moarref and Jovanović [21] and Moarref and Jovanović [22] to design controllers to suppress the onset of tur-

bulence in a channel, and to achieve turbulent drag reduction in a channel, respectively. Similar tools were also used in Ran et al. [26] to design spanwise-periodic riblets with the objective of reducing drag in a turbulent channel. For a thorough overview of frequency-domain methods for the analysis of fluids flows we refer to the review paper by Jovanović [13] and references therein.

2. Frequential reachability and observability Gramians

In continuous-time balanced truncation for time-periodic systems, one seeks a time-periodic reduced-order model by first identifying a change of coordinates that simultaneously diagonalizes the time-periodic reachability and observability Gramians, which will be defined below. While the balancing procedure will be discussed in detail in section 6, for now, it suffices to say that computing the Gramians (or their factorization) is the most computationally expensive step. In this section we therefore focus on the computation of the Gramians and we show that, similarly to the linear time-invariant case, these can be defined in the frequency domain. We also show that the frequency-domain representation of the Gramians is well defined even when the underlying dynamics are unstable; more specifically, the frequential Gramians satisfy a pair of differential Lyapunov equations.

2.1. Preliminaries

We begin by considering a linear time-periodic system with state $x(t) \in \mathbb{R}^N$, control input $u(t) \in \mathbb{R}^M$, and output $y(t) \in \mathbb{R}^Q$

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} x(t) &= A(t)x(t) + B(t)u(t) \\
y(t) &= C(t)x(t),
\end{aligned}
\tag{1}
$$

where the linear operators $A(t)$, $B(t)$ and $C(t)$ are all periodic with period T (i.e., $A(t) = A(t + T)$). The system (1) arises in fluid mechanics when the Navier-Stokes equations are linearized about a T-periodic solution. In incompressible flow, the state $x(t)$ may be taken as the divergence-free velocity field at the cell faces (or cell centers) of a computational grid, the forcing term $B(t)u(t)$ may be understood as a volumetric or boundary input and the output $y(t)$ could be some measured quantity (e.g., the velocity at some desired physical location in the flow). Usually, the operators $B(t)$ and $C(t)$ are time-invariant (e.g., if the control input and the measured output are located at some fixed physical coordinate), but here we include time dependence for the sake of generality.

In order to derive the frequency-domain representation of the reachability and observability Gramians associated with (1), it is notationally convenient to first diagonalize (1) via a Floquet change of coordinates [10]. In particular, by Floquet's theorem, there exists a possibly complex T-periodic change of coordinates $x(t) = V(t)z(t)$ such that

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} z(t) &= J z(t) + \underbrace{W(t)^* B(t)}_{\widetilde{B}(t)} u(t) \\
y(t) &= \underbrace{C(t)V(t)}_{\widetilde{C}(t)} z(t),
\end{aligned}
\tag{2}
$$

where $W(t)^* V(t) = I$ for all t and J is a diagonal time-invariant matrix containing the Floquet exponents associated with (1). Here, W^* denotes the Hermitian transpose of W. It is easy to verify that the periodic orbit is stable if and only if all the Floquet exponents lie in the left-half plane, and it is unstable otherwise.

2.2. Stable dynamics

In this subsection we assume that (1) is stable in the sense just described. The reachability and observability Gramians associated with the diagonalized dynamics (2) may be defined as

$$
G_R(t_0, t) = \int_{t_0}^{t} e^{J(t-\tau)} \widetilde{B}(\tau) \widetilde{B}(\tau)^* e^{J^*(t-\tau)} \mathrm{d}\tau
\tag{3}
$$

$$
G_O(t, t_f) = \int_{t}^{t_f} e^{J^*(\tau-t)} \widetilde{C}(\tau)^* \widetilde{C}(\tau) e^{J(\tau-t)} \mathrm{d}\tau.
\tag{4}
$$

It can be shown that as $t_0 \to -\infty$ and $t_f \to +\infty$, the Gramians $G_R(t_0, t)$ and $G_O(t, t_f)$ are periodic functions of t with period T. This is a well-known result that can be illustrated by considering the forward and adjoint differential Lyapunov equations below

$$
\frac{\mathrm{d}}{\mathrm{d}t} P(t) = J P(t) + P(t) J^* + \widetilde{B}(t) \widetilde{B}(t)^*
\tag{5}
$$

$$
-\frac{\mathrm{d}}{\mathrm{d}t} Q(t) = J^* Q(t) + Q(t) J + \widetilde{C}(t)^* \widetilde{C}(t).
\tag{6}
$$

It can be readily checked that the solution $\boldsymbol{P}(t)$ of (5) may be written as

$$\boldsymbol{P}(t) = \underbrace{e^{\boldsymbol{J}(t-t_0)}\boldsymbol{P}(t_0)e^{\boldsymbol{J}^*(t-t_0)}}_{\text{initial cond. response}} + \underbrace{\boldsymbol{G}_R(t_0,t)}_{\text{forced resp.}}, \tag{7}$$

where $\boldsymbol{P}(t_0)$ is the initial condition at time $t = t_0$. By Theorem 20 in Bolzern and Colaneri [4], equation (5) admits a unique positive-definite (for all times) T-periodic solution if (2) is stable and controllable. This solution may be understood as the long-time response of (5) to the external forcing $\widetilde{\boldsymbol{B}}(t)\widetilde{\boldsymbol{B}}(t)^*$. In particular, if (1) is stable, it is clear that for $t \gg t_0$ the initial condition response in (7) will go to zero and we will be left with the forced response $\boldsymbol{P}(t) = \boldsymbol{G}_R(t_0,t)$. A similar argument holds for the adjoint differential Lyapunov equation (6) for $t \ll t_f$.

Now that we have established the relationship between the Gramians and the periodic solution of the corresponding differential Lyapunov equations, we can seek this solution in the frequency domain. In particular, since the solution is periodic with period T, we can write

$$\boldsymbol{P}(t) = \sum_{k\in\mathbb{Z}} \boldsymbol{P}_k e^{ik\omega t}, \quad \boldsymbol{Q}(t) = \sum_{k\in\mathbb{Z}} \boldsymbol{Q}_k e^{ik\omega t}, \quad \omega = \frac{2\pi}{T}. \tag{8}$$

Moreover, since $\widetilde{\boldsymbol{B}}(t)$ and $\widetilde{\boldsymbol{C}}(t)$ are also T-periodic, they can be written in a Fourier series analogous to the ones in (8). Substitution into the corresponding Lyapunov equations leads, for a fixed integer k, to

$$(-ik\omega\boldsymbol{I} + \boldsymbol{J})\boldsymbol{P}_k + \boldsymbol{P}_k\boldsymbol{J}^* + \sum_{l\in\mathbb{Z}} \widetilde{\boldsymbol{B}}_{k+l}\widetilde{\boldsymbol{B}}_l^* = 0 \tag{9}$$

$$(ik\omega\boldsymbol{I} + \boldsymbol{J}^*)\boldsymbol{Q}_k + \boldsymbol{Q}_k\boldsymbol{J} + \sum_{l\in\mathbb{Z}} \widetilde{\boldsymbol{C}}_{-l-k}^*\widetilde{\boldsymbol{C}}_{-l} = 0. \tag{10}$$

It is easy to verify (see, e.g., Godunov [11]) that the solution of the two algebraic Sylvester equations (9) and (10) is given by

$$\boldsymbol{P}_k = \frac{1}{2\pi}\int_{-\infty}^{\infty} (i\gamma\boldsymbol{I} - (-ik\omega\boldsymbol{I} + \boldsymbol{J}))^{-1} \sum_{l\in\mathbb{Z}} \widetilde{\boldsymbol{B}}_{k+l}\widetilde{\boldsymbol{B}}_l^* \left(-i\gamma\boldsymbol{I} - \boldsymbol{J}^*\right)^{-1} \, d\gamma \tag{11}$$

$$\boldsymbol{Q}_k = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left(-i\gamma\boldsymbol{I} - (ik\omega\boldsymbol{I} + \boldsymbol{J}^*)\right)^{-1} \sum_{l\in\mathbb{Z}} \widetilde{\boldsymbol{C}}_{-l-k}^*\widetilde{\boldsymbol{C}}_{-l}(i\gamma\boldsymbol{I} - \boldsymbol{J})^{-1} \, d\gamma. \tag{12}$$

It can be shown that the above integrals converge for all k as long as no Floquet exponent (i.e., eigenvalue of $\boldsymbol{J}$) lies on the imaginary axis. This implies that $\boldsymbol{P}_k$ and $\boldsymbol{Q}_k$ are well-defined also when one or more of the Floquet exponents lie in the open right-half plane (i.e., when the periodic dynamics are unstable). In the next subsection, we show that when the dynamics are unstable, the Gramians in (11) and (12) satisfy corresponding differential Lyapunov equations.

2.3. Unstable dynamics

In this subsection we assume that one or more of the eigenvalues of $\boldsymbol{J}$ lie in the open right-half plane, so that the periodic dynamics are unstable. We henceforth let $\mathcal{P}_s$ and $\mathcal{P}_u$ denote the projections onto the stable and unstable eigenspaces of $\boldsymbol{J}$. Notice that since $\boldsymbol{J}$ is a diagonal matrix, $\mathcal{P}_s$ and $\mathcal{P}_u$ are orthogonal projections (and, in fact, diagonal themselves). We can then state the following result.

Proposition 1. *Suppose that no eigenvalue of $\boldsymbol{J}$ lies on the imaginary axis. Then the Fourier coefficients $\boldsymbol{P}_k$ and $\boldsymbol{Q}_k$ defined in* (11) *and* (12) *satisfy the following Sylvester equations,*

$$(-ik\omega\boldsymbol{I} + \boldsymbol{J})\boldsymbol{P}_k + \boldsymbol{P}_k\boldsymbol{J}^* + \mathcal{P}_s\sum_{l\in\mathbb{Z}}\widetilde{\boldsymbol{B}}_{k+l}\widetilde{\boldsymbol{B}}_l^*\mathcal{P}_s - \mathcal{P}_u\sum_{l\in\mathbb{Z}}\widetilde{\boldsymbol{B}}_{k+l}\widetilde{\boldsymbol{B}}_l^*\mathcal{P}_u = 0 \tag{13}$$

$$(ik\omega\boldsymbol{I} + \boldsymbol{J}^*)\boldsymbol{Q}_k + \boldsymbol{Q}_k\boldsymbol{J} + \mathcal{P}_s\sum_{l\in\mathbb{Z}}\widetilde{\boldsymbol{C}}_{-l-k}^*\widetilde{\boldsymbol{C}}_{-l}\mathcal{P}_s - \mathcal{P}_u\sum_{l\in\mathbb{Z}}\widetilde{\boldsymbol{C}}_{-l-k}^*\widetilde{\boldsymbol{C}}_{-l}\mathcal{P}_u = 0. \tag{14}$$

Via inverse Fourier transform, it follows that the T-periodic Gramians $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ satisfy the differential Lyapunov equations below

$$\frac{d}{dt}\boldsymbol{P}(t) = \boldsymbol{J}\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{J}^* + \mathcal{P}_s\widetilde{\boldsymbol{B}}(t)\widetilde{\boldsymbol{B}}(t)^*\mathcal{P}_s - \mathcal{P}_u\widetilde{\boldsymbol{B}}(t)\widetilde{\boldsymbol{B}}(t)^*\mathcal{P}_u \tag{15}$$

$$-\frac{d}{dt}\boldsymbol{Q}(t) = \boldsymbol{J}^*\boldsymbol{Q}(t) + \boldsymbol{Q}(t)\boldsymbol{J} + \mathcal{P}_s\widetilde{\boldsymbol{C}}(t)^*\widetilde{\boldsymbol{C}}(t)\mathcal{P}_s - \mathcal{P}_u\widetilde{\boldsymbol{C}}(t)^*\widetilde{\boldsymbol{C}}(t)\mathcal{P}_u. \tag{16}$$

Proof. If all the Floquet exponents lie in the left-half plane, then $\mathcal{P}_u = \boldsymbol{0}$ and $\mathcal{P}_s = \boldsymbol{I}$, and equations (13) and (14) agree with (9) and (10), respectively. In the general case when $\mathcal{P}_u \neq \boldsymbol{0}$, the proof is analogous to the linear time-invariant case in Section 10.2 of Godunov [11]. The second part of the proposition follows immediately via inverse Fourier transform. $\square$

We have therefore established that the frequency-domain representation of the Gramians is well-defined for both stable and unstable dynamics. By contrast, the time-domain integral representation of the Gramians is well-defined only if the dynamics are stable; otherwise, the initial condition response in (7) would blow up for $t \gg t_0$. In the next section we will take steps to make practical use of the results discussed in this section.

3. Towards an efficient algorithm to compute the Gramians

In this section we address two main issues. First and foremost, in balanced truncation we are not interested in explicitly computing the Fourier coefficients of the Gramians. Instead, we would like to evaluate the Gramians at a desired time $t \in [0, T]$. Conveniently, we show in the upcoming subsection that the Gramians at any time t can be computed as an outer product of frequency-domain quantities. In the subsequent subsections, we address the second issue. Namely, we would like to compute the Gramians in the physical coordinates $\boldsymbol{x}(t)$. While the diagonalized Floquet coordinates $\boldsymbol{z}(t)$ have proven useful to understand the structure of the Gramians, they are not well-suited for computation. Specifically, for high-dimensional systems, it is generally infeasible to compute the diagonalizing Floquet transformation that maps $\boldsymbol{x}(t)$ coordinates to $\boldsymbol{z}(t)$ coordinates.

3.1. Gramians as an outer product

For the time being, we still work in Floquet coordinates $\boldsymbol{z}(t)$, and we begin by showing that the Gramians $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ may be computed at specific times t as an outer product of frequency-domain factors. This is convenient from a computational standpoint, since integrals written as outer products lend themselves to straightforward numerical quadrature. The content of this subsection may therefore be understood as a first step towards developing an algorithm to compute the Gramians using frequency-domain variables.

Proposition 2. *The T-periodic reachability and observability Gramians $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ can be written as*

$$\boldsymbol{P}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Z}(\gamma, t) \boldsymbol{Z}(\gamma, t)^* \, \mathrm{d}\gamma, \quad \boldsymbol{Q}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Y}(\gamma, t) \boldsymbol{Y}(\gamma, t)^* \, \mathrm{d}\gamma, \tag{17}$$

where

$$\boldsymbol{Z}(\gamma, t) = \sum_{m \in \mathbb{Z}} \boldsymbol{Z}_m(\gamma) e^{im\omega t} = \sum_{m \in \mathbb{Z}} \left((i\gamma \boldsymbol{I} - (-im\omega \boldsymbol{I} + \boldsymbol{J}))^{-1} \widetilde{\boldsymbol{B}}_m \right) e^{im\omega t} \tag{18}$$

$$\boldsymbol{Y}(\gamma, t) = \sum_{m \in \mathbb{Z}} \boldsymbol{Y}_m(\gamma) e^{im\omega t} = \sum_{m \in \mathbb{Z}} \left((-i\gamma \boldsymbol{I} - (im\omega \boldsymbol{I} + \boldsymbol{J}^*))^{-1} \widetilde{\boldsymbol{C}}_{-m}^* \right) e^{im\omega t}. \tag{19}$$

Proof. The proof relies on the linearity of the Sylvester equations (13) and (14), so that it can be shown that the Fourier coefficients $\boldsymbol{P}_k$ and $\boldsymbol{Q}_k$ of the Gramians can be written as linear combination of quantities that satisfy Sylvester equations similar to (13) and (14). Details can be found in Appendix A.1. $\quad\square$

We have therefore written the reachability Gramian $\boldsymbol{P}(t)$ as an outer product of a matrix-valued function $\boldsymbol{Z}(\gamma, t)$ and its complex conjugate transpose. It is now easy to see that for any fixed time $t \in [0, T]$, one can estimate $\boldsymbol{P}(t)$ by numerically evaluating the integral in (17) via quadrature. The most computationally-intensive part of evaluating this integral is the computation of $\boldsymbol{Z}_m(\gamma)$. Remarkably, however, this computation only needs to be performed once: one may compute and store $\boldsymbol{Z}_m(\gamma)$ for every m and γ, and then, for every desired time $t \in [0, T]$, we simply have to rotate $\boldsymbol{Z}_m(\gamma)$ by $e^{im\omega t}$ and evaluate the integral.

3.2. Gramians in physical coordinates and connection with the harmonic resolvent

We are now ready to transition to physical coordinates $\boldsymbol{x}(t)$, which are well-suited for computation. Given the Gramians $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ in the Floquet coordinates $\boldsymbol{z}(t)$, one may verify that the Gramians in the original $\boldsymbol{x}(t)$ coordinates are given by

$$\boldsymbol{P}_{\boldsymbol{x}}(t) = \boldsymbol{V}(t) \boldsymbol{P}(t) \boldsymbol{V}(t)^*, \quad \boldsymbol{Q}_{\boldsymbol{x}}(t) = \boldsymbol{W}(t) \boldsymbol{Q}(t) \boldsymbol{W}(t)^*, \tag{20}$$

where, as before, the T-periodic matrices $\boldsymbol{V}(t)$ and $\boldsymbol{W}(t)$ define the Floquet change of coordinates in (2). Using the integral representation of $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$ in Proposition 2, it readily follows that

$$\boldsymbol{P}_{\boldsymbol{x}}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t) \boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t)^* \, \mathrm{d}\gamma, \; \boldsymbol{Q}_{\boldsymbol{x}}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Y}_{\boldsymbol{x}}(\gamma, t) \boldsymbol{Y}_{\boldsymbol{x}}(\gamma, t)^* \, \mathrm{d}\gamma, \tag{21}$$

where

$$\boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t) := \boldsymbol{V}(t) \boldsymbol{Z}(\gamma, t) = \sum_{k \in \mathbb{Z}} \underbrace{\sum_{m \in \mathbb{Z}} \boldsymbol{V}_{k-m} \boldsymbol{Z}_m(\gamma)}_{:=\boldsymbol{Z}_{\boldsymbol{x},k}(\gamma)} e^{ik\omega t}, \tag{22}$$

$$Y_x(\gamma, t) := W(t)Y(\gamma, t) = \sum_{k \in \mathbb{Z}} \underbrace{\sum_{m \in \mathbb{Z}} W_{k-m} Y_m(\gamma)}_{:=Y_{x,k}(\gamma)} e^{ik\omega t}. \tag{23}$$

Using the definition of $Z_m(\gamma)$ and $Y_m(\gamma)$ in Proposition 2, and the definition of $\widetilde{B}(t)$ and $\widetilde{C}(t)$ in (2), we may further expand $Z_{x,k}(\gamma)$ and $Y_{x,k}(\gamma)$ as follows

$$Z_{x,k}(\gamma) = \sum_{m \in \mathbb{Z}} V_{k-m} (i\gamma I - (-im\omega I + J))^{-1} \sum_{j \in \mathbb{Z}} W_{j-m}^* B_j \tag{24}$$

$$Y_{x,k}(\gamma) = \sum_{m \in \mathbb{Z}} W_{k-m} \left(-i\gamma I - (im\omega I + J^*)\right)^{-1} \sum_{j \in \mathbb{Z}} V_{j-m}^* C_{-j}^*. \tag{25}$$

We now show that $Z_{x,k}(\gamma)$ and $Y_{x,k}(\gamma)$ can be computed using the harmonic resolvent operator.

We begin with a short derivation of the harmonic resolvent operator. More details may be found in Padovan and Rowley [25]. Starting from (1) we write the state vector $x(t)$ as

$$x(t) = e^{i\gamma t} \sum_{k \in \mathbb{Z}} x_{k+\gamma} e^{ik\omega t}, \quad \gamma \in [0, \omega/2]. \tag{26}$$

The signal above is known as an exponentially modulated periodic (EMP) signal, where the T-periodic component inside the sum is modulated by the complex exponential $e^{i\gamma t}$. It is well-known that EMPs are the appropriate class of signals for the analysis of time-periodic signals (see, e.g., Johnson [12] or Wereley [32]). For later reference, we observe that the signal $x(t)$ is a sum of Fourier modes with frequencies in the set $\Omega_\gamma = \gamma + \omega\mathbb{Z}$, where + denotes element-wise addition. Since all the linear operators in (1) are periodic with period T, they may be written in a Fourier series analogous to (8). Then, writing $u(t)$ as an EMP, formula (1) may be written in the frequency domain as

$$i(\gamma + k\omega)x_{k+\gamma} = \sum_{j \in \mathbb{Z}} A_{k-j} x_{j+\gamma} + \sum_{j \in \mathbb{Z}} B_{k-j} u_{j+\gamma}$$

$$y_{k+\gamma} = \sum_{j \in \mathbb{Z}} C_{k-j} x_{j+\gamma}. \tag{27}$$

Letting $\hat{x}_\gamma = (\dots, x_{-1+\gamma}, x_\gamma, x_{1+\gamma}, \dots)$ denote an infinite-dimensional vector that contains all the coefficients of the EMP signal (26), we can define the infinite-dimensional linear operator T as

$$\left[T\hat{x}_\gamma\right]_k = -ik\omega x_{k+\gamma} + \sum_{j \in \mathbb{Z}} A_{k-j} x_{j+\gamma}. \tag{28}$$

Notice that T is independent of γ, as it depends only on ω and on the Fourier coefficients of $A(t)$. From this definition and from formula (27), it follows that $x_{k+\gamma}$ is given by

$$x_{k+\gamma} = \sum_{j,l \in \mathbb{Z}} \underbrace{\left[(i\gamma I - T)^{-1}\right]_{k,j}}_{H_{k,j}(\gamma)} B_{j-l} u_{l+\gamma} \tag{29}$$

where the operator

$$H(\gamma) = (i\gamma I - T)^{-1} \tag{30}$$

is known as the harmonic resolvent operator evaluated at γ, and $H_{k,j}(\gamma)$ is the block of $H(\gamma)$ that maps inputs at frequency $(\gamma + j\omega) \in \Omega_\gamma$ to outputs at frequency $(\gamma + k\omega) \in \Omega_\gamma$. We can now state the desired result.

Proposition 3. *The Fourier coefficients $Z_{x,k}(\gamma)$ and $Y_{x,k}(\gamma)$ of the Gramian factors in formulas (24) and (25) may be written in terms of the harmonic resolvent $H(\gamma)$ as follows*

$$Z_{x,k}(\gamma) = \sum_{j \in \mathbb{Z}} H_{k,j}(\gamma) B_j, \quad Y_{x,k}(\gamma) = \sum_{j \in \mathbb{Z}} [H(\gamma)^*]_{k,j} [C^*]_j. \tag{31}$$

Proof. Using a Floquet change of coordinates, it can be shown that

$$H_{k,j}(\gamma) = \sum_{m \in \mathbb{Z}} V_{k-m} (i\gamma I - (-im\omega I + J))^{-1} W_{j-m}^*, \tag{32}$$

and this concludes the proof. More details can be found in Appendix A.2. $\square$

Formula (21) and Proposition 3 give us the necessary building blocks for a practical algorithm to compute the Gramians at desired times $t \in [0, T)$. In particular, the proposition shows us how to evaluate the integrals in (21) using the easily accessible harmonic resolvent operator.

4. An algorithm to compute a factorization of the Gramians

In this section we highlight some of the features of the harmonic resolvent $\boldsymbol{H}(\gamma)$ that can be exploited to minimize computational cost. We then provide an explicit algorithm to compute the factors $\boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t)$ and $\boldsymbol{Y}_{\boldsymbol{x}}(\gamma, t)$.

4.1. Symmetries in the harmonic resolvent operator

We henceforth focus on the reachability Gramian $\boldsymbol{P}(t)$, since the computation of the observability Gramian $\boldsymbol{Q}(t)$ can be carried out in a similar fashion. For a given time $t \in [0, T)$, we can estimate the integral (21) as follows

$$\boldsymbol{P}_{\boldsymbol{x}}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t) \boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t)^* \mathrm{d}\gamma \approx \frac{1}{2\pi} \sum_{l \in \mathbb{N}} \xi_l \boldsymbol{Z}_{\boldsymbol{x}}(\gamma_l, t) \boldsymbol{Z}_{\boldsymbol{x}}(\gamma_l, t)^* \tag{33}$$

where ξ_l are quadrature coefficients, γ_l are discrete samples over the interval $(-\infty, \infty)$ and

$$\boldsymbol{Z}_{\boldsymbol{x}}(\gamma_l, t) = \sum_{k \in \mathbb{Z}} \boldsymbol{Z}_{\boldsymbol{x},k}(\gamma_l) e^{ik\omega t} = \sum_{k \in \mathbb{Z}} \left(\sum_{j \in \mathbb{Z}} \boldsymbol{H}_{k,j}(\gamma_l) \boldsymbol{B}_j \right) e^{ik\omega t} \in \mathbb{C}^{N \times M} \tag{34}$$

by Proposition 3. In order to evaluate $\boldsymbol{Z}_{\boldsymbol{x}}(\gamma, t)$ at different times, we can simply compute and store the Fourier coefficients $\boldsymbol{Z}_{\boldsymbol{x},k}(\gamma)$ and then rotate them using the complex exponential. This is advantageous, since computing $\boldsymbol{Z}_{\boldsymbol{x},k}(\gamma)$ is an expensive operation that could easily become computationally intractable if it had to be performed multiple times. In particular, computing matrix-matrix products of the form $\boldsymbol{H}_{k,j}(\gamma) \boldsymbol{B}_j$ requires inverting the operator $i\gamma \boldsymbol{I} - \boldsymbol{T}$ (see formula (29)). The cost of solving these linear systems is dominated either by the computation of a complete factorization of $i\gamma \boldsymbol{I} - \boldsymbol{T}$ (e.g., LU decomposition), or by the computation of a preconditioner to assist the convergence of iterative solvers such as GMRES. This cost could be intractable if it had to be sustained for many values γ in the interval $(-\infty, \infty)$. Fortunately, we now show that the factorization (or computation of a preconditioner) needs to be performed for only a few values of γ in the interval $[0, \omega/2]$.

We begin with the following proposition, which states that $\boldsymbol{Z}(\alpha, t)$ for any $\alpha \in (-\infty, \infty)$ may be computed using the harmonic resolvent $\boldsymbol{H}(\gamma)$ evaluated at $\gamma \in (-\omega/2, \omega/2]$.

Proposition 4. *For any $\alpha \in \mathbb{R}$, there exists an integer m such that $\gamma = \alpha - m\omega \in (-\omega/2, \omega/2]$ and*

$$\boldsymbol{Z}_{\boldsymbol{x}}(\alpha, t) = \sum_{k,j \in \mathbb{Z}} \boldsymbol{H}_{k,j}(\gamma) \boldsymbol{B}_{j-m} e^{i(k-m)\omega t}. \tag{35}$$

Proof. Here we present an intuitive reason why this result holds. Recall that $\boldsymbol{H}(\alpha)$ maps inputs over the frequency set $\Omega_\alpha = \alpha + \omega\mathbb{Z}$ to outputs over the same set Ω_α. Clearly, if $\alpha = \gamma + m\omega$ (for an integer m), then $\Omega_\alpha = \Omega_\gamma = \gamma + \omega\mathbb{Z}$. So, in order to compute $\boldsymbol{Z}_{\boldsymbol{x}}(\alpha, t)$, we can use the harmonic resolvent $\boldsymbol{H}(\gamma)$ evaluated at γ. The rigorous proof is in Appendix A.3. $\square$

A second observation that we can make to reduce the computational cost stems from the real-valued nature of the dynamics in (1). In particular, it can be shown that for every γ, we have $\boldsymbol{Z}(-\gamma, t) = \overline{\boldsymbol{Z}(\gamma, t)}$, where the overline denotes complex conjugation. Thus, the desired Gramian $\boldsymbol{P}(t)$ may be approximated as

$$\boldsymbol{P}_{\boldsymbol{x}}(t) \approx \frac{1}{\pi} \sum_{l \in \mathbb{N}} \xi_l \left[\boldsymbol{Z}_{\boldsymbol{x},\mathrm{r}}(\alpha_l, t) \boldsymbol{Z}_{\boldsymbol{x},\mathrm{r}}(\alpha_l, t)^* + \boldsymbol{Z}_{\boldsymbol{x},\mathrm{i}}(\alpha_l, t) \boldsymbol{Z}_{\boldsymbol{x},\mathrm{i}}(\alpha_l, t)^* \right] c_l, \tag{36}$$

where $\alpha_l \geq 0$, the subscripts "r" and "i" denote the real and imaginary parts of $\boldsymbol{Z}_{\boldsymbol{x}}(\alpha_l, t)$, and $c_l = 1$ for $\alpha_l > 0$ and $1/2$ for $\alpha_l = 0$. In other words, the integral can be approximated by considering only positive values $\alpha \in [0, \infty)$. Putting together this observation and the result from Proposition 4, it is clear that we only need to factorize (or compute a preconditioner for) $\boldsymbol{H}(\gamma)$ at values $\gamma \in [0, \infty) \cap (-\omega/2, \omega/2] = [0, \omega/2]$.

4.2. Practical algorithm to compute the Gramian factors

As a first step for practical implementation, we need to truncate the Fourier representations of the periodic components of the dynamics. In particular, we truncate $\boldsymbol{A}(t)$, $\boldsymbol{B}(t)$, and $\boldsymbol{C}(t)$ at frequency r_b, so that, for instance, we write

$$\boldsymbol{A}(t) = \sum_{k=-r_b}^{r_b} \boldsymbol{A}_k e^{ik\omega t}. \tag{37}$$

Similarly, we truncate the EMP signal in (26) as follows

$$\boldsymbol{x}(t) = e^{i\gamma t} \sum_{k=-r}^{r} \boldsymbol{x}_k e^{ik\omega t}, \tag{38}$$

where we take $r \geq r_b$. It follows that $\boldsymbol{T}$ is a square matrix with size $(2r+1)N$ and structure shown below,

$$
T = \begin{bmatrix}
\ddots & \ddots & \ddots & \ddots & & & & \\
\ddots & R_{-2} & A_{-1} & A_{-2} & \ddots & & & \\
\ddots & A_1 & R_{-1} & A_{-1} & A_{-2} & \ddots & & \\
\ddots & A_2 & A_1 & R_0 & A_{-1} & A_{-2} & \ddots & \\
& \ddots & A_2 & A_1 & R_1 & A_{-1} & \ddots & \\
& & \ddots & A_2 & A_1 & R_2 & \ddots & \\
& & & \ddots & \ddots & \ddots & &
\end{bmatrix},
\tag{39}
$$

where $R_k = \left(-ik\omega I + A_0\right) \in \mathbb{C}^{N\times N}$. Recalling that $i\gamma I - T$ acts on vectors $\hat{x}_\gamma$, we henceforth use the notation $[\hat{x}_\gamma]_k$ to indicate the portion of the vector $\hat{x}_\gamma$ that gets multiplied by kth block-column of T. (For clarity, the kth block-column is the one containing the block R_k.) We can now observe that for fixed m and γ, the quantity $Z_{x,k}(\alpha) := \sum_j H_{k,j}(\gamma) B_{j-m}$ (with $\alpha = \gamma - m\omega$, see Proposition 4) may be computed simultaneously for all $k \in \{-r, \dots, r\}$. For example, taking $\gamma = 0$ and $m = 1$, the quantities $Z_{x,k}(\omega) = \sum_j H_{k,j}(0) B_{j-1}$ satisfy

$$
-\underbrace{\begin{bmatrix}
\ddots & \ddots & & & \\
\ddots & R_{-1} & A_{-1} & & \\
& A_1 & R_0 & A_{-1} & \\
& & A_1 & R_1 & \ddots \\
& & & \ddots & \ddots
\end{bmatrix}}_{i\gamma I - T = -T}
\underbrace{\begin{bmatrix}
\vdots \\
Z_{x,-1}(\omega) \\
Z_{x,0}(\omega) \\
Z_{x,1}(\omega) \\
\vdots
\end{bmatrix}}_{\hat{Z}_{x,\gamma}^{(m)} = \hat{Z}_{x,0}^{(1)}}
=
\underbrace{\begin{bmatrix}
\vdots \\
B_{-2} \\
B_{-1} \\
B_0 \\
\vdots
\end{bmatrix}}_{\hat{B}^{(m)} = \hat{B}^{(1)}}.
\tag{40}
$$

Notice that the matrix $\hat{B}^{(m)}$ is defined such that $[\hat{B}^{(m)}]_k = B_{k-m}$. In general then, given fixed m and γ, we compute $Z_{x,k}(\alpha) = \sum_j H_{k,j}(\gamma) B_{j-m}$ as follows:

$$
\text{solve} \quad (i\gamma I - T)\hat{Z}_{x,\gamma}^{(m)} = \hat{B}^{(m)}, \quad \text{extract} \quad Z_{x,k}(\alpha) := [\hat{Z}_{x,\gamma}^{(m)}]_k.
\tag{41}
$$

Algorithm 1 Compute factor $\widetilde{Z}_x(t_n)$ for $t_n \in [0, T)$.

Input: Matrix T and discrete points $\gamma_l \in [0, \omega/2]$ with $l \in \{1, 2, \dots, L\}$
Output: Matrix $\widetilde{Z}_x(t)$ at time $t \in [0, T)$

Part I: Compute $Z_{x,k}(\gamma_l + m\omega) := \sum_j H_{k,j}(\gamma) B_{j-m}$ **for all** γ_l **and** m

---

1: Initialize matrix $X \in \mathbb{C}^{(2r+1)N \times ((L-2)(2r+1)+2(r+1))M}$
2: **for** $l \in \{1, 2, \dots, L\}$ **do**
3:     Compute factorization (or preconditioner) of $i\gamma_l I - T$
4:     **if** $\gamma_l \neq 0$ and $\gamma_l \neq \omega/2$ **then**
5:         Range $= \{-r, \dots, 0, \dots, r\}$
6:     **else**
7:         Range $= \{0, \dots, r\}$
8:     **for** $m \in$ Range **do**
9:         Solve $(i\gamma_l I - T)\hat{Z}_{x,\gamma_l}^{(m)} = \hat{B}^{(m)}$, where $[\hat{Z}_{x,\gamma_l}^{(m)}]_k = Z_{x,k}(\gamma_l + m\omega)$
10:      Store $\hat{Z}_{x,\gamma_l}^{(m)}$ in X
11: **Return:** Matrix X

Part II: Compute $\widetilde{Z}_x(t_n)$ **at some desired time** $t_n \in [0, T)$

12: Initialize $\widetilde{Z}_x(t_n) \in \mathbb{R}^{N \times 2((L-2)(2r+1)+2(r+1))M}$
13: **for** $l \in \{1, 2, \dots, L\}$ **do**
14:     **if** $\gamma_l \neq 0$ and $\gamma_l \neq \omega/2$ **then**
15:         Range $= \{-r, \dots, 0, \dots, r\}$
16:     **else**
17:         Range $= \{0, \dots, r\}$
18:     **if** $\gamma_l = 0$ **then**
19:         $c_l = 1/2$
20:     **else**
21:         $c_l = 1$
22:     **for** $m \in$ Range **do**
23:         Extract the component $\hat{Z}_{\gamma_l}^{(m)}$ from X
24:         Compute $Z_{x,\gamma_l}^{(m)}(t_n) \leftarrow \sum_{k \in \mathbb{Z}_x} [\hat{Z}_{\gamma_l}^{(m)}]_k e^{i(k-m)\omega t_n}$
25:         Store $\sqrt{\frac{c_l \xi_l}{\pi}} \operatorname{Real}(Z_{x,\gamma_l}^{(m)}(t_n))$ and $\sqrt{\frac{c_l \xi_l}{\pi}} \operatorname{Imag}(Z_{x,\gamma_l}^{(m)}(t_n))$ into $\widetilde{Z}_x(t_n)$
26: **Return:** Factor $\widetilde{Z}_x(t_n)$

We are now ready to present an algorithm to compute the Gramian factors. In particular, Algorithm 1 will output a matrix $\widetilde{Z}_x(t)$ with columns

$$\frac{1}{\sqrt{\pi}}\left\{\sqrt{c_l\xi_l}\boldsymbol{Z}_{\boldsymbol{x},\mathrm{r}}(\alpha_l,t),\sqrt{c_l\xi_l}\boldsymbol{Z}_{\boldsymbol{x},\mathrm{i}}(\alpha_l,t)\right\},\quad \alpha_l\in[0,\infty),\tag{42}$$

so that, per equation (36), we have $\boldsymbol{P}_{\boldsymbol{x}}(t)\approx\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)^*$.

In the first part of the algorithm, we compute the frequency-domain factors and store them. This is the most computationally intensive part of the algorithm, as we need to solve several linear systems of size $(2r+1)N$. In the second part, we simply rotate the previously computed factors using the complex exponential, and evaluate the factor $\boldsymbol{Z}_{\boldsymbol{x}}(t)$ at the desired time $t\in[0,T]$. This part of the algorithm is virtually free of cost compared to the first part. We close this section by observing that the factors $\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)$ may be computed using Algorithm 1 by replacing $i\gamma_l\boldsymbol{I}-\boldsymbol{T}$ with its complex conjugate transpose and by replacing $\boldsymbol{B}$ with $\boldsymbol{C}^*$. The observability Gramian at any time t may then be evaluated via quadrature as $\boldsymbol{Q}_{\boldsymbol{x}}(t)\approx\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)^*$.

5. Computational considerations

In this section we provide guidelines that may be helpful in determining whether the Gramians should be computed in the frequency domain or in the time domain, for a particular application. We use a three-dimensional toy model to demonstrate the use and cost of frequency-domain and time-domain algorithms to compute the Gramians.

5.1. Heuristic guidelines

Here, we consider the reachability Gramian $\boldsymbol{P}_{\boldsymbol{x}}(t)$, since analogous logic applies to the observability Gramian $\boldsymbol{Q}_{\boldsymbol{x}}(t)$. In order to appreciate the benefits and drawbacks of using frequential Gramians, it is instructive to understand how the Gramians would be computed in the time domain. For a *stable* system (1), we recall that

$$\boldsymbol{P}_{\boldsymbol{x}}(t)=\boldsymbol{P}_{\boldsymbol{x}}(t+T):=\lim_{n\to\infty}\boldsymbol{G}_{R,\boldsymbol{x}}(0,nT+t),\tag{43}$$

where n is a positive integer and

$$\boldsymbol{G}_{R,\boldsymbol{x}}(0,s)=\int_0^s\boldsymbol{F}(s,\tau)\boldsymbol{B}(\tau)\boldsymbol{B}(\tau)^*\boldsymbol{F}(s,\tau)^*\mathrm{d}\tau.\tag{44}$$

Here, $\boldsymbol{F}(s,\tau)$ denotes the fundamental solution of (1); i.e., $\mathrm{d}/\mathrm{d}s(\boldsymbol{F}(s,\tau))=\boldsymbol{A}(s)\boldsymbol{F}(s,\tau)$ with $\boldsymbol{F}(\tau,\tau)=\boldsymbol{I}$. For a fixed $\tau\leq s$, the quantity

$$\boldsymbol{x}_\tau(s):=\boldsymbol{F}(s,\tau)\boldsymbol{B}(\tau)\in\mathbb{R}^{N\times M}\tag{45}$$

is the time-τ impulse response of (1), which can be computed numerically by solving

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}_\tau^{(j)}(t)=\boldsymbol{A}(t)\boldsymbol{x}_\tau^{(j)}(t),\quad \boldsymbol{x}_\tau^{(j)}(\tau)=\boldsymbol{B}_j(\tau),\quad j\in\{1,2,\dots,M\},\tag{46}$$

from $t=\tau$ to $t=s$. Formula (43) naturally lends itself to numerical quadrature, so that $\boldsymbol{P}_{\boldsymbol{x}}(t)$ may be approximated as $\boldsymbol{P}_{\boldsymbol{x}}(t)\approx\lim_{n\to\infty}\boldsymbol{X}(nT+t)\boldsymbol{X}(nT+t)^*$, where $\boldsymbol{X}(s)$ is given below

$$\boldsymbol{X}(s)=\left[\sqrt{\xi_1}\boldsymbol{x}_{\tau_1}(s),\sqrt{\xi_2}\boldsymbol{x}_{\tau_2}(s),\dots,\sqrt{\xi_S}\boldsymbol{x}_{\tau_S}(s)\right]\in\mathbb{C}^{N\times MS},\quad \tau_l\in[0,s],\tag{47}$$

and ξ_l are quadrature coefficients. By contrast, we recall from part II of Algorithm 1 that the Gramian $\boldsymbol{P}_{\boldsymbol{x}}(t)$ may be approximated in the frequency domain as $\boldsymbol{P}_{\boldsymbol{x}}(t)\approx\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)^*$, where $\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)$ has size $N\times 2MI$ and I is equal to the number of discrete α_l in equation (36). The factor of 2 comes from the fact that the real and imaginary parts of $\boldsymbol{Z}_{\boldsymbol{x}}(\alpha,t)$ in equation (36) are stored separately.

With this information at hand, we see that evaluating $\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)$ at some fixed time t requires solving MI algebraic systems of equations. By contrast, evaluating $\boldsymbol{X}(s)$ at some fixed time $s=nT+t$ requires solving MS initial-value problems (46) in the time domain. The question we ask is, when is it convenient to compute $\boldsymbol{P}_{\boldsymbol{x}}(t)$ in the frequency domain, and when is it convenient to compute it in the time domain? Although it is virtually impossible to provide a precise operations count for the two methods, we can still provide guidelines that the user may find useful. For simplicity, let us assume that $O(I)=O(S)$. This can usually be taken to be the case in practice. The fundamental difference between the two methods is that one requires time stepping, while the other does not. Thus, although the number of required impulse responses MS and required linear solves MI is comparable, the time-domain method is fundamentally limited by *(i)* numerical stability constraints associated with time stepping (i.e., the time step in (46) might have to be small, depending on the specific properties of the time-stepper) and *(ii)* physical transients that time-steppers are forced to follow. In particular, if the dynamics exhibit slowly-decaying transients, then $s=nT+t$ must be taken very large (see formula (43)), and the cost of computing the Gramian using the time domain increases significantly. By contrast, computing the Gramian factors in the frequency domain requires solving algebraic systems of equations that do not suffer from the drawbacks of time-stepping methods. Another often overlooked aspect of time steppers is that they are inherently sequential in time, so that any type of computational parallelism can only be spatial (e.g., distributing the degrees of freedom of (1) across multiple processors). On the other hand, algebraic systems of equations lend themselves to massive space-time parallelism, so that a higher number of processors can be deployed to accelerate the computations in the frequency domain.

Unfortunately, the benefits of the frequency domain are not free of cost. The main drawback is that the N-dimensional time-periodic dynamics are "lifted" into a higher-dimensional space of size $N(2r+1)$. Therefore, solving the desired algebraic equations

requires inverting the operator $i\gamma I - T$. The main issue therefore lies in computing a factorization of $i\gamma - T$, or a suitable preconditioner (although, fortunately, this only has to be done for a few values $\gamma \in [0, \omega/2)$, as explained in the previous section). Depending on the nature and size of the underlying problem, it may be possible to compute an LU decomposition of $i\gamma I - T$ using parallelized libraries such as MUMPS [2], or alternatively, one may use out-of-the-box preconditioners. Given the structure of T, we recommend the use of Block-Jacobi as a starting point. However, there may be problems whose size is such that an LU factorization cannot be performed, and whose structure is such that a Block-Jacobi preconditioner does not work particularly well. Identifying an efficient preconditioner tailored to the quasi block-Toeplitz structure of the matrix $i\gamma I - T$ remains an open question and the subject of future work.

As far as storage is concerned, the time-domain approach is more efficient. Although both approaches yield Gramian factors of comparable size, part I of Algorithm 1 shows that the matrix X of size $N(2r+1) \times MI$ needs to be held in memory. Thus, the memory burden of the frequency-domain approach is $O(r)$ higher than the time-domain method.

Given this discussion, it becomes clear that the choice of algorithm is heavily dependent on the nature of the underlying dynamics. In general, we recommend the use of the frequency domain for systems with state N of moderate size $O(10^5)$ or less, so that the size of T remains below $O(10^7)$ and an LU decomposition of $i\gamma I - T$ can be computed. If the size of the system is much higher than $O(10^5)$, but the structure of T is predominantly diagonal, then the frequency domain remains a feasible option, with out-of-the-box preconditioners such as Block-Jacobi assisting the convergence of Krylov solvers. Systems where the structure of T is dominantly diagonal (see, e.g., (39)) are systems where $\|A_{k>0}\| \ll \|A_0\|$, where A_k is the kth Fourier coefficient of the operator $A(t)$. We also recommend the use of the frequency domain if the underlying system exhibits slowly-decaying transients, since, as explained, these will inevitably drive up the cost of computing the Gramians in the time domain. Finally, the frequency domain is also well-suited for unstable systems, while the time-domain procedure described herein cannot be applied since the limit in (43) does not exist. Theoretically, this issue could be solved via a splitting of the stable and unstable Floquet eigenspaces, but this comes at the price of higher computational cost.

5.2. Application to a three-dimensional toy model

In this section we implement the frequency-domain and time-domain algorithms discussed so far to compute the Gramians for a simple system of three differential equations. All the scripts used to generate these results are publicly available in the GitHub repository https://github.com/albertopadovan/Frequential_Gramians. The system we consider is governed by the following differential equations

$$\dot{x} = \mu x - y - \alpha xz - \beta xy + 0.1u \tag{48}$$

$$\dot{y} = x + \mu y - \alpha yz + \beta x^2 + 0.1u \tag{49}$$

$$\dot{z} = -\alpha z + \alpha\left(x^2 + y^2\right) \tag{50}$$

where $\dot{x} = dx/dt$, $\mu = \alpha = \beta = 1/5$, $u(t)$ is an external input chosen as $u(t) = \sin(2\omega t)$ and $\omega = \sqrt{1 - \beta^2\mu/\alpha}$ is the natural frequency of the system. A detailed discussion of the dynamics of this system is presented in section 3 in Padovan et al. [24]. The external input u is chosen to be non zero to set the phase of the limit cycle and remove the neutrally stable direction of phase shift that would naturally appear otherwise. The limit cycle induced by the external forcing u is shown in Fig. 1a and colored according to $\|dx/dt\|$, where $x = (x, y, z)$ is the state vector. Upon linearization of the dynamics (48)–(50) about the periodic limit cycle shown in Fig. 1a, we obtain a linear time-periodic system of the form

$$\frac{d}{dt}x' = A(t)x' + Bu', \tag{51}$$

where $x' = (x', y', z')$ denotes the perturbed state about the limit cycle and $B = (0.1, 0.1, 0)$ is the input matrix of size 3×1. A representative initial-condition response of the linearized dynamics is shown in Fig. 1b with the intent of demonstrating that this system exhibits slowly-decaying transients: perturbations decay to zero after 40 to 50 fundamental periods $T = 2\pi/\omega$. We shall see momentarily that this slow decay severely increases the cost of computing the Gramians in the time domain.

We now compute the reachability Gramian P_x at time $t = 0$ using three different methods. The first method computes the Gramian using the time-domain approach described in section 5.1. The second method computes the Gramian using the frequency-domain factors, but *without* exploiting the symmetries in the harmonic resolvent operator discussed in section 4.1. That is, we factorize/invert the matrix $i\gamma_l I - T$ for every $\gamma_l \in [0, \infty)$. Finally, the third method uses Algorithm 1. That is, we leverage the symmetries in the harmonic resolvent operator and we factorize $i\gamma_l I - T$ only for values $\gamma_l \in [0, \omega/2)$. Throughout this section, we take $r_b = r = 10$ in equations (37) and (38), so that T has rows (and column) size $3 \times (2r+1) = 63$. When computing the Gramian in the time domain, we approximate the integral (44) by taking $s = 40T$, and we discretize the integration domain using 600 points (i.e., 15 points per period T). Since we have only one input (i.e., u' is a scalar), it follows that we have to compute a total of 600 impulse responses from times $\tau_l \in [0, 40T]$ to $s = 40T$. When computing the Gramian using the frequential factors, we discretize the integration domain $[0, \infty)$ with 610 equally-spaced points with $\Delta\gamma = (\omega/2)/30$, regardless of whether we leverage the harmonic resolvent symmetries or not. Once again, since we only have one input, we have to solve 610 linear systems in the frequency domain. The numerical experiments reported herein were performed on a MacBook Pro with M2 chips running Python 3.9.13. Time-stepping was performed using `solve_ivp` from the `scipy.integrate` library, while matrix factorizations were performed using `scipy.linalg`.
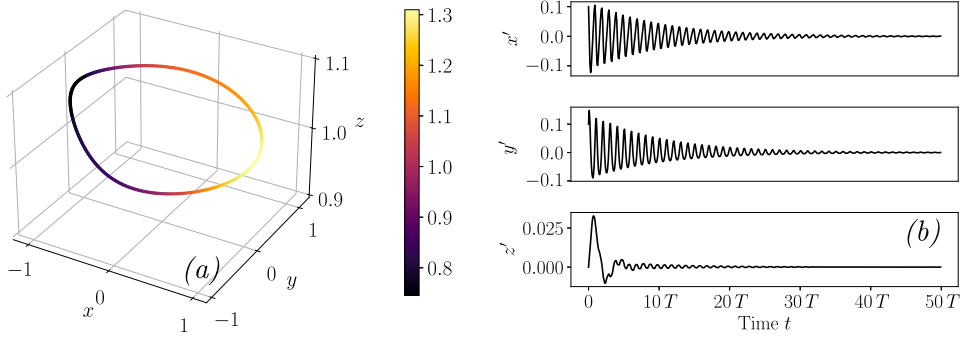
Fig. 1. *(a)* Limit-cycle solution of equations (48)–(50) in response to $u = \sin(2\omega t)$ (the colorbar shows the 2-norm of the time-derivative of the solution) and, *(b)* initial condition response of the linearized dynamics (51) with initial condition $\boldsymbol{x}'(0) = (0.1, 0.1, 0.0)$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Table 1
Comparison of the three approaches used to compute the reachability Gramian at time $t = 0$. The percent error is defined in formula (52).

	Wall-clock time [s]	Percent error
Time domain	30.50	7.5×10^{-3}
Frequency domain (no symmetries)	0.060	2.0×10^{-3}
Frequency domain (Algorithm 1)	0.019	2.0×10^{-3}

In Table 1 we report the execution time of the three approaches, along with the percent error computed against a ground truth reference. The ground truth reference $\boldsymbol{P}_{\boldsymbol{x},\text{ref}}(0)$ was computed by harmonic-balancing the differential Lyapunov equation (5) in physical coordinates (thus solving a frequency-domain algebraic Lyapunov equation), and the percent error is defined as

$$e = 100 \times \frac{\text{Trace}\left((\boldsymbol{P}_{\boldsymbol{x}} - \boldsymbol{P}_{\boldsymbol{x},\text{ref}})^T (\boldsymbol{P}_{\boldsymbol{x}} - \boldsymbol{P}_{\boldsymbol{x},\text{ref}})\right)}{\text{Trace}\left(\boldsymbol{P}_{\boldsymbol{x},\text{ref}}^T \boldsymbol{P}_{\boldsymbol{x},\text{ref}}\right)}, \tag{52}$$

where $\boldsymbol{P}_{\boldsymbol{x}}$ is the Gramian (at time $t = 0$) computed by any one of the three methods that we discuss. As we can see in the table, the time-domain approach is by far the most computationally intensive. This is likely due to the fact that the system at hand exhibits slowly decaying transients that force us to run very long impulse responses. By contrast, the "naive" frequency-domain approach is roughly 500 times faster, while Algorithm 1 achieves a speed-up of an additional factor of 3. The additional computational gain obtained using Algorithm 1 is due to the fact that we compute fewer matrix factorizations. In fact, we perform 10 times fewer factorizations that in the naive frequency-domain implementation. For larger problems, where the matrix factorization is the computational bottleneck, we can expect the additional speed-up achieved by Algorithm 1 to scale (approximately) linearly with the decrease in matrix factorizations. Finally, we also observe that the error associated with the frequency-domain approaches is approximately 3 to 4 times lower than the error given by the time-domain approach. We expect the error of the time-domain approach to decrease as we increase the integration final time s in (44), and as we refine the sampling of the integration interval τ.

6. Continuous-time balanced truncation

In this section we describe the continuous-time balanced truncation approach for model reduction. Given a dynamical system of the form (1), balanced truncation seeks a *continuously differentiable* periodic change of coordinates $\boldsymbol{x}(t) = \boldsymbol{\Phi}(t)\boldsymbol{q}(t)$, with $\boldsymbol{\Psi}(t)^*\boldsymbol{\Phi}(t) = \boldsymbol{I} \in \mathbb{R}^{N \times N}$, such that the $\boldsymbol{q}(t)$-coordinate Gramians

$$\boldsymbol{P}_{\boldsymbol{q}}(t) = \boldsymbol{\Psi}(t)^* \boldsymbol{P}_{\boldsymbol{x}}(t) \boldsymbol{\Psi}(t), \quad \boldsymbol{Q}_{\boldsymbol{q}}(t) = \boldsymbol{\Phi}(t)^* \boldsymbol{Q}_{\boldsymbol{x}}(t) \boldsymbol{\Phi}(t) \tag{53}$$

are equal and diagonal. In other words,

$$\boldsymbol{P}_{\boldsymbol{q}}(t) = \boldsymbol{Q}_{\boldsymbol{q}}(t) = \boldsymbol{\Sigma}(t) = \text{diag}(\sigma_1(t), \sigma_2(t), \dots, \sigma_N(t)). \tag{54}$$

We begin by illustrating the balancing scheme, and then we address some of the subtleties associated with the time-varying nature of the problem. Given a factorization of the Gramians, e.g.,

$$\boldsymbol{P}_{\boldsymbol{x}}(t) = \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t) \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)^*, \quad \boldsymbol{Q}_{\boldsymbol{x}}(t) = \widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t) \widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)^*, \tag{55}$$

where the factors can be chosen to be T-periodic (see, e.g., the factors described in the previous sections), the first step in computing the balancing change of coordinates is to compute the singular value decomposition (SVD)

$$\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)^* \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t) = \boldsymbol{U}(t)\boldsymbol{\Sigma}(t)\boldsymbol{V}(t)^* \tag{56}$$

at all times $t \in [0, T]$. Then, so long as $\boldsymbol{\Sigma}(t)^{-1}$ exists for all t, the desired matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$ are given by

$$\boldsymbol{\Phi}(t) = \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)\boldsymbol{V}(t)\boldsymbol{\Sigma}(t)^{-1/2}, \quad \boldsymbol{\Psi}(t) = \widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)\boldsymbol{U}(t)\boldsymbol{\Sigma}(t)^{-1/2}, \tag{57}$$

where it can be easily checked that $\boldsymbol{\Psi}(t)^*\boldsymbol{\Phi}(t) = \boldsymbol{I}$ for all times.

The first subtlety stems from the fact that we require the matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$ to be continuously differentiable. As a consequence, while in linear time-invariant systems the (time-invariant) Hankel singular values σ_i may be arranged in descending order, in the time-varying setting it may not be possible to enforce this arrangement for all times. In fact, the differentiability requirement can cause the time-periodic $\sigma_i(t)$ to cross each other for different i. This seemingly innocuous difference may lead to ambiguities when trying to determine a global (i.e., for all times) truncation rank $r \ll N$ to assemble a reduced-order model. In fact, it is theoretically possible that the smallest singular value at time $t = 0$ becomes the largest at a later time. Fortunately, while the singular values can certainly coalesce in practice, the crossing is usually localized, so that the aforementioned pathological behavior is uncommon. For the sake of completeness, we observe that this issue can be addressed by allowing for the reduced-order system to have time-varying dimensions. While this is a natural thing to do in discrete-time setting [31], the possibility of time-varying state dimensions in continuous-time has been explored by Sandberg and Rantzer [28] at the price of introducing discontinuities in the measured output $\boldsymbol{y}(t)$.

Another subtlety that arises in continuous-time balanced truncation is the fact that even though the full-order model (1) is periodic with period T, the balanced model may have period mT for some integer $m \geq 1$. This is a direct consequence of the fact that smooth decompositions of T-periodic matrices (in this case $\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t)^* \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t)$) may yield factors (see equation (56)) whose period is larger than T. A thorough discussion on smoothness and periodicity of some matrix decompositions may be found in Chern and Dieci [5]. These two difficulties are inherently tied to the balancing procedure and they cannot be avoided. Fortunately, however, they can be addressed in a straightforward fashion within our framework. An algorithm is provided below.

Algorithm 2 Compute matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$.

Input: Matrix $\boldsymbol{T}$, discrete samples $\gamma_l \in [0, \omega/2]$, reduced-order model rank r, expected period mT of $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$, discrete time samples $t_n \in [0, mT]$
Output: Matrices $\boldsymbol{\Phi}(t_n), \boldsymbol{\Psi}(t_n) \in \mathbb{R}^{N \times r}$ at time instances $t_n \in [0, mT]$

1: Compute and store matrix $\boldsymbol{X}$ using part I of Algorithm 1
2: **for** $t_n \in [0, mT)$ **do**
3:     Compute $\widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t_n)$ and $\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t_n)$ using part II of Algorithm 1
4:     Compute the SVD of $\widetilde{\boldsymbol{Y}}_{\boldsymbol{x}}(t_n)^* \widetilde{\boldsymbol{Z}}_{\boldsymbol{x}}(t_n) = \boldsymbol{U}(t_n)\boldsymbol{\Sigma}(t_n)\boldsymbol{V}(t_n)^*$ as in (56)
5:     Truncate the SVD factors at rank r
6:     **if** $t_n > 0$ **then**
7:         Order the SVD factors so that they are continuously differentiable
        with respect to time t: permute the columns of the factors so that
        $\boldsymbol{U}(t_{n-1})^T \boldsymbol{U}(t_n) \approx \boldsymbol{I}$.
8:     Compute $\boldsymbol{\Phi}(t_n)$ and $\boldsymbol{\Psi}(t_n)$ as in (57)

The "if statement" in the algorithm may be understood as a mode-tracking step. Given a sufficiently finely sampled time interval, it is reasonable to expect the SVD factors $\boldsymbol{U}(t_n)$ and $\boldsymbol{V}(t_n)$ to be well-aligned with the factors $\boldsymbol{U}(t_{n-1})$ and $\boldsymbol{V}(t_{n-1})$. Comparing the mode alignment at neighboring time instances allows us to detect any crossing of the singular values and to keep the factors continuously differentiable across the entire interval $[0, mT)$. For completeness, it is worth observing that different mode-tracking logic can be implemented. For example, this could be done using an approach similar to the dynamical low rank approximation described in Koch and Lubich [15] and in Lubich and Oseledets [18]. In Algorithm 2, the time interval $[0, mT)$ itself is given as an input to the algorithm. Unfortunately, there is no practical a-priori way of determining what the period of the factors will be [5], so the appropriate value mT will be problem dependent. Fortunately, it is inexpensive to try different values of m (or even to choose m sufficiently large and then identify the minimal period), since the computationally-intensive part of Algorithm 2 consists in computing the matrix $\boldsymbol{X}$ via part I of Algorithm 1.

Given the matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$ of period mT, the desired mT-periodic r-dimensional reduced model is obtained by substituting $\boldsymbol{x}(t) = \boldsymbol{\Phi}(t)\boldsymbol{q}(t)$ into (1) and left-multiplying by $\boldsymbol{\Psi}(t)^*$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{q}(t) = \underbrace{\boldsymbol{\Psi}(t)^* \left(\boldsymbol{A}(t)\boldsymbol{\Phi}(t) - \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\Phi}(t) \right)}_{\boldsymbol{A}_r(t)} \boldsymbol{q}(t) + \underbrace{\boldsymbol{\Psi}(t)^* \boldsymbol{B}(t)}_{\boldsymbol{B}_r(t)} \boldsymbol{u}(t)$$

$$\boldsymbol{y}(t) = \underbrace{\boldsymbol{C}(t)\boldsymbol{\Phi}(t)}_{\boldsymbol{C}_r(t)} \boldsymbol{q}(t). \tag{58}$$

This is a mT-periodic linear system whose size $r \ll N$ is suitable for control and estimation.

7. Application to an axisymmetric jet

In this section we demonstrate the balancing Algorithms 1 and 2 on a periodically-forced incompressible axisymmetric jet at two different Reynolds numbers, $Re = 1250$ and $Re = 1500$. At $Re = 1250$, the flow admits a stable time-periodic solution characterized
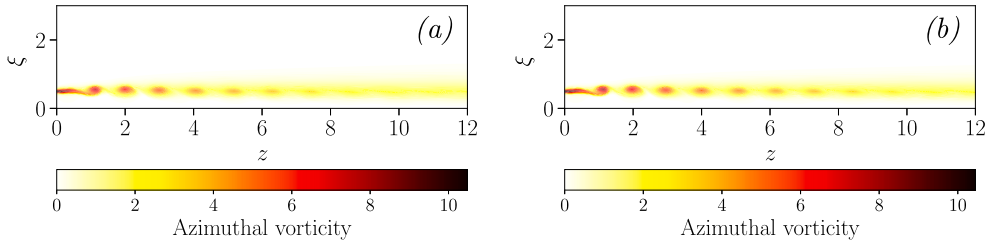
Fig. 2. Vorticity snapshots from the T-periodic solution at time $t = 0$ for *(a)* Reynolds number $Re = 1250$ and *(b)* Reynolds number $Re = 1500$.

by unpaired vortex rings. However, this solution is extremely sensitive to subharmonic perturbations, so that any small perturbation will cause neighboring vortex rings to pair and merge. Here, we compute a reduced-order model and we design a disturbance-rejection feedback controller to delay and mitigate the pairing phenomenon. At $Re = 1500$, the flow admits an *unstable* time-periodic solution, also characterized by unpaired vortex rings. Given the unstable nature of the solution, however, the flow will naturally depart from the unstable unpaired configuration and it will settle onto a different periodic orbit characterized by paired rings. In this case, we compute a reduced-order model and we use it to design a stabilizing feedback controller. These two cases demonstrate the effectiveness of Algorithms 1 and 2 at delivering a reduced-order model both when the underlying dynamics are stable and when they are unstable.

7.1. Flow description and numerical setup

We begin by providing a brief description of the governing equations. Throughout, velocities are non-dimensionalized by the jet centerline velocity U_0 and lengths are non-dimensionalized by the jet diameter D_0, so that we may define the Reynolds number $Re = U_0 D_0 / \nu$, where ν is the kinematic viscosity of the fluid. The flow is governed by the incompressible Navier-Stokes equation along with the continuity equation over the spatial domain $D = \{(z, \xi) | z \in [0, L_z], \xi \in [0, L_\xi]\}$, with $L_z = 15$ and $L_\xi = 4$. In particular, given the (dimensionless) axial velocity u, the radial velocity v and the pressure p, we have

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial z} + v \frac{\partial u}{\partial \xi} = -\frac{\partial p}{\partial z} + \frac{1}{Re} \left(\frac{1}{\xi} \frac{\partial}{\partial \xi} \left(\xi \frac{\partial u}{\partial \xi} \right) + \frac{\partial^2 u}{\partial z^2} \right) \tag{59a}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial z} + v \frac{\partial v}{\partial \xi} = -\frac{\partial p}{\partial \xi} + \frac{1}{Re} \left(\frac{1}{\xi} \frac{\partial}{\partial \xi} \left(\xi \frac{\partial v}{\partial \xi} \right) - \frac{v}{\xi^2} + \frac{\partial^2 v}{\partial z^2} \right) \tag{59b}$$

$$\frac{\partial u}{\partial z} + \frac{1}{\xi} \frac{\partial (\xi v)}{\partial \xi} = 0. \tag{59c}$$

At the centerline $\xi = 0$ we impose axisymmetric boundary conditions, at the outflow and at the top boundary we impose a zero normal gradient boundary condition on both velocity components, and at the inflow we consider the axial velocity profile

$$u(\xi, z = 0, t) = g(\xi)(1 + A \cos \omega t), \tag{60}$$

where A is the non-dimensional forcing amplitude, ω is the forcing frequency, and

$$g(\xi) = \frac{1}{2} \left\{ 1 - \tanh \left[\frac{1}{4\theta_0} \left(\xi - \frac{1}{4\xi} \right) \right] \right\}. \tag{61}$$

The parameter θ_0 may be understood as a non-dimensional vorticity thickness of the incoming profile. The spatial domain is discretized on a fully-staggered grid using second-order finite differences, except for the advective term, which is treated using a third-order upwind-biased scheme. Given the fully-staggered nature of the grid, we do not require explicit pressure boundary conditions. Throughout, we work on a grid of size $N_z \times N_\xi = 600 \times 200$, we fix $A = 0.05$, $\theta_0 = 0.025$ and $\omega = 2\pi 0.6$, and we consider two different Reynolds numbers $Re = 1250$ and $Re = 1500$. For this choice of parameters, it is shown by Shaabani-Ardali et al. [30] via a Floquet stability analysis that the T-periodic solution is stable at $Re = 1250$ and unstable at $Re = 1500$. For both cases, we compute the T-periodic solution via time-stepping of the Navier-Stokes equations (59) augmented with the time-delay feedback technique described in Shaabani-Ardali et al. [29]. This technique is necessary to compute unstable solutions via time-stepping (as in the $Re = 1500$ case), but it can also be used to suppress transients in stable configurations (e.g., the $Re = 1250$ case) thereby accelerating the convergence to the desired post-transient solution. Representative snapshots from the two solutions are shown in Fig. 2.

7.2. Actuator and sensor configurations

We now provide details concerning the model-reduction procedure and we also describe the actuator and sensor configurations. Given the T-periodic solution $X = (U, V)$ of the Navier-Stokes equations (59), the perturbed velocity field $x' = (u', v')$ and the perturbed pressure p', the linearized dynamics are governed by the equations

$$\frac{\partial u'}{\partial t} + B_z(x', X) = -\frac{\partial p'}{\partial z} + \frac{1}{Re} \left(\frac{1}{\xi} \frac{\partial}{\partial \xi} \left(\xi \frac{\partial u'}{\partial \xi} \right) + \frac{\partial^2 u'}{\partial z^2} \right) \tag{62a}$$

$$\frac{\partial v'}{\partial t} + B_\xi(\boldsymbol{x}', \boldsymbol{X}) = -\frac{\partial p'}{\partial \xi} + \frac{1}{Re}\left(\frac{1}{\xi}\frac{\partial}{\partial \xi}\left(\xi\frac{\partial v'}{\partial \xi}\right) - \frac{v'}{\xi^2} + \frac{\partial^2 v'}{\partial z^2}\right) \tag{62b}$$

$$\frac{\partial u'}{\partial z} + \frac{1}{\xi}\frac{\partial\left(\xi v'\right)}{\partial \xi} = 0, \tag{62c}$$

where

$$B_z(\boldsymbol{x}', \boldsymbol{X}) = u'\frac{\partial U}{\partial z} + U\frac{\partial u'}{\partial z} + v'\frac{\partial U}{\partial \xi} + V\frac{\partial u'}{\partial \xi}$$

$$B_\xi(\boldsymbol{x}', \boldsymbol{X}) = u'\frac{\partial V}{\partial z} + U\frac{\partial v'}{\partial z} + v'\frac{\partial V}{\partial \xi} + V\frac{\partial v'}{\partial \xi}.$$

The boundary conditions on the perturbed velocity are analogous to those imposed on the full velocity field, except for the inflow where we impose zero Dirichlet boundary conditions on both velocity components. Upon removal of the pressure p' via a Poisson equation and letting $\boldsymbol{x}'(t)$ denote the spatially-discretized velocity at the cell faces of the computational grid, the system (62) may be written as a linear time-periodic system in standard form

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}'(t) = \boldsymbol{A}(t)\boldsymbol{x}'(t), \quad \boldsymbol{A}(t) = \boldsymbol{A}(t+T). \tag{63}$$

Given our grid size, the state vector $\boldsymbol{x}'(t)$ will have size $N = 2N_z N_\xi = 2.4 \times 10^5$.

At this point, we need to specify how the control input will enter the dynamics. This is our first design choice, and we decide to actuate the flow through an axial velocity body force localized in the proximity of $z_c = 1$ and $\xi_c = 0.5$, with magnitude

$$\exp\left[-\frac{1}{\theta_0}\left((z - z_c)^2 + (\xi - \xi_c)^2\right)\right] u(t), \tag{64}$$

where $u(t)$ is our control input. Thus, the matrix $\boldsymbol{B}$ in equation (1) is a single column of height N. Notice that here, the matrix $\boldsymbol{B}$ is time-invariant, which means that the control input always enters the dynamics at the same physical location. We choose the location (z_c, ξ_c) near the jet nozzle, since this is where one might be able to place an actuator in practice. Additional insight into the actuator placement was also obtained from the sensitivity analysis in Padovan and Rowley [25], where the authors showed that the flow is most sensitive to axial velocity perturbations in the proximity of the nozzle. In general, in fact, actuators should be placed in regions of physical space where the flow is most "observable" (or, more intuitively, most sensitive to perturbations), so that a "small" control input can induce a "large" effect on the flow. It is also worth reporting that the exact location of the actuator did not seem to affect the performance of the closed-loop system, so long as it lied within the sensitivity region identified in Padovan and Rowley [25].

The second design choice concerns sensor placement. Throughout, we choose to measure the axial velocity at four different locations with radial coordinate $\xi_c = 0.5$ and axial coordinates $z_c = \{1.5, 2.5, 5, 6\}$. This yields a time-invariant matrix $\boldsymbol{C} \in \mathbb{R}^{4 \times N}$, where each row is given by a spatial profile analogous to the one in (64). In choosing the sensor locations we considered the following. First, one or more sensors should be placed in proximity of the actuator in order to mitigate the detrimental effect of delays between the input and the measured response. Second, one or more sensors should be placed farther downstream since this is the location of the vortex pairing phenomenon that we wish to suppress. In general, for linear systems as the one considered herein, sensors should be placed in regions of physical space where the flow is most "controllable". In the case of the jet, these are the regions of physical space where the flow develops large-scale deviations from the underlying base flow. While there exist systematic/optimal ways to place sensors for a given system, here we placed them based on the previously described heuristics. Some tests (not shown here) indicated that the closed-loop system defined in the upcoming section was not very sensitive to the exact location of the four sensors. However, we found, for example, that using fewer than three or four sensors led to poor performance of the observer and, consequently, to the inability of properly controlling the flow.

Given our $\boldsymbol{B}$ and $\boldsymbol{C}$ matrices, we henceforth work with the input-output system below

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}'(t) &= \boldsymbol{A}(t)\boldsymbol{x}'(t) + \boldsymbol{B}u(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}'(t),\end{aligned} \tag{65}$$

where $u(t) \in \mathbb{R}$ is our control input (which will be determined by an appropriate feedback law) and $\boldsymbol{y}(t) \in \mathbb{R}^4$ is the measured output.

In order to compute the balancing transformation matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$ using Algorithms 1 and 2, we need to assemble the matrix $\boldsymbol{T}$ (39) associated with the linearized dynamics (65). In particular, we truncate the Fourier representation of $\boldsymbol{A}(t)$ at $r_b = 4$ harmonics of the fundamental frequency ω (see formula (37)) and we truncate the EMP representation of the state $\boldsymbol{x}'(t)$ at $r = 6$ harmonics (see formula (38)). Thus, the matrix $\boldsymbol{T}$ will have size $(2r + 1)N = 3.12 \times 10^6$. Algorithm 1 is implemented in a PETSc-based solver run on the Princeton Tiger Cluster, and the linear solvers in the algorithm are preconditioned with PETSc's built-in Block-Jocobi preconditioner. The interval $[0, \omega/2]$ in Algorithm 1 is discretized using 11 equally-spaced points $\gamma_l \in [0, \omega/2]$. After computing the balancing transformation matrices $\boldsymbol{\Phi}(t)$ and $\boldsymbol{\Psi}(t)$, we can explicitly assemble a reduced-order model of the form (58), where in this case the matrices $\boldsymbol{B}$ and $\boldsymbol{C}$ are time-invariant.

7.3. Feedback controller and state estimator design

Given a reduced-order model of the form (58) with reduced state $\boldsymbol{q}(t) \in \mathbb{R}^r$, we can now approach the task of designing a feedback controller to modify the full-order dynamics. We design the feedback controller using the linear quadratic regulator (LQR) approach

for linear time-periodic systems. A thorough overview of the LQR problem for time-periodic systems is given in Wereley [32], while rigorous results on the existence and uniqueness of a periodic feedback law may be found in Bittanti et al. [3]. Simply put, given the mT-periodic linear system (58), the LQR method yields a mT-periodic feedback control law $u(t) = -\boldsymbol{K}(t)\boldsymbol{q}(t)$ by solving the optimization problem

$$\min_{\boldsymbol{K}(t)} \quad \mathcal{J}_{LQR} = \int_0^\infty \left(\boldsymbol{q}(t)^* \boldsymbol{Q}_q(t)\boldsymbol{q}(t) + u(t)^2 \right) \mathrm{d}t, \tag{66}$$

subject to the dynamics in (58). Here, $\boldsymbol{Q}_q(t)$ is a positive-semidefinite $r \times r$ matrix that quantifies the relative importance of driving the states to zero, versus maintaining small control inputs. Our choice of $\boldsymbol{Q}_q(t)$ is informed by the analysis carried out in section 5C of Padovan and Rowley [25]. In particular, for $Re = 1250$, we demonstrated that the pairing phenomenon that we wish to suppress is driven exclusively by a $2T$-periodic mode denoted $\boldsymbol{\xi}(t)$. In order to suppress (or mitigate) vortex pairing, we therefore need to design a controller that rejects perturbations whose projection onto $\boldsymbol{\xi}(t)$ is non-zero. Given the full-order state $\boldsymbol{x}'(t) = \boldsymbol{\Phi}(t)\boldsymbol{q}(t)$, and letting $\boldsymbol{\xi}(t)$ be normalized such that

$$\frac{1}{2T} \int_0^{2T} \boldsymbol{\xi}(t)^* \boldsymbol{\xi}(t)\,\mathrm{d}t = \sum_{k \in \mathbb{Z}} \hat{\boldsymbol{\xi}}_k^* \hat{\boldsymbol{\xi}}_k = 1, \tag{67}$$

the projection of $\boldsymbol{x}'(t)$ onto $\boldsymbol{\xi}(t)$ is given by $\boldsymbol{\xi}(t)^*\boldsymbol{\Phi}(t)\boldsymbol{q}(t)$. This information may be encoded into the LQR problem by choosing $\boldsymbol{Q}_q(t)$ as follows,

$$\boldsymbol{Q}_q(t) = \alpha \left(\boldsymbol{\Phi}(t)^* \boldsymbol{\xi}(t)\boldsymbol{\xi}(t)^* \boldsymbol{\Phi}(t) \right), \tag{68}$$

where α is a positive scalar. The matrix $\boldsymbol{Q}_q(t)$ now contributes to the cost function $\mathcal{J}_{LQR}$ by measuring the projection of the full-state onto the "most dangerous" mode. Consequently, the resulting optimal feedback law will try to change the dynamics by minimizing the projection of the state onto $\boldsymbol{\xi}(t)$. The same rationale was applied in the $Re = 1500$ case.

In order to implement the feedback law discussed above, it is necessary to design a state estimator (or observer), which, given the available sensor measurements, computes a state estimate $\widetilde{\boldsymbol{q}}(t)$. The desired control input will then be given by $u(t) = -\boldsymbol{K}(t)\widetilde{\boldsymbol{q}}(t)$. Here, we design an observer using the linear quadratic estimator approach, which assumes that the reduced state $\boldsymbol{q}(t)$ and the measured output $\boldsymbol{y}(t)$ are corrupted by Gaussian noise. More specifically, we suppose that $\boldsymbol{q}(t)$ and $\boldsymbol{y}(t)$ are governed by

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{q}(t) &= \boldsymbol{A}_r(t)\boldsymbol{q}(t) + \boldsymbol{B}_r(t)u(t) + \overline{\boldsymbol{B}}_r(t)d(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}_r(t)\boldsymbol{q}(t) + \boldsymbol{n}(t), \end{aligned} \tag{69}$$

where the disturbance $d(t)$ and the sensor noise $\boldsymbol{n}(t)$ are zero-mean Gaussian processes with covariance $\mathbb{E}[d(t)d(\tau)] = \beta\,\delta(t - \tau)$ and $\mathbb{E}[\boldsymbol{n}(t)\boldsymbol{n}(\tau)^*] = \boldsymbol{Q}_n\delta(t - \tau)$, respectively. The linear operator $\overline{\boldsymbol{B}}_r(t)$ is mT-periodic and it is chosen by the user to model how the disturbance $d(t)$ enters the dynamics. We will elaborate on this choice shortly. It may be shown that the optimal state estimate $\widetilde{\boldsymbol{q}}(t)$ is governed by the dynamics below

$$\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\boldsymbol{q}}(t) = \left(\boldsymbol{A}_r(t) - \boldsymbol{L}(t)\boldsymbol{C}_r(t) \right) \widetilde{\boldsymbol{q}}(t) + \boldsymbol{B}_r(t)u(t) + \boldsymbol{L}(t)\boldsymbol{y}(t), \tag{70}$$

where the mT-periodic matrix $\boldsymbol{L}(t)$ is chosen to minimize the expected estimation error

$$\lim_{t \to \infty} \mathbb{E}\left[\left\| \boldsymbol{q}(t) - \widetilde{\boldsymbol{q}}(t) \right\| \right].$$

In our implementation, we choose $\boldsymbol{Q}_n$ to be a diagonal matrix with entries

$$[\boldsymbol{Q}_n]_{i,i} = \frac{\max_t |\boldsymbol{C}_r(t)|}{\max_t |\boldsymbol{C}_{r,i}(t)|}, \tag{71}$$

where $\boldsymbol{C}_{r,i}$ denotes the ith row of the output matrix $\boldsymbol{C}_r(t)$. This ensures that the resulting estimator responds equally strongly (or weakly) to changes in each measured output. Finally, we design the matrix $\overline{\boldsymbol{B}}_r(t)$ once again by leveraging the results from Padovan and Rowley [25]. According to that analysis, the only external disturbances that have a measurable effect on the flow are those that align with the aforementioned mode $\boldsymbol{\xi}(t)$. Therefore, we choose

$$\overline{\boldsymbol{B}}_r(t) = \boldsymbol{\Psi}(t)^* \boldsymbol{\xi}(t). \tag{72}$$

That is, we model the disturbances that enter the dynamics via the "most dangerous" mode and we disregard all the other ones. A block diagram of the observer-based feedback configuration is shown in Fig. 3. Before moving forward, it is worth observing that the optimal estimation problem we just described is the dual of the LQR problem used for the controller design, and the solution to both problems can be obtained by solving the corresponding differential Riccati equations. Conveniently, these differential Riccati equations can be converted into equivalent frequency-domain *algebraic* Riccati equations (see section 5.2 of Wereley [32]), which can be solved easily with the `lqr` command in MATLAB or `solve_continuous_are` from `scipy.linalg` in Python.
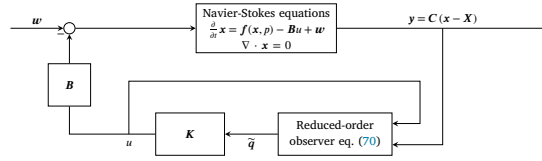
Fig. 3. Block diagram for the observer-based feedback configuration. The plant (labelled "Navier-Stokes equations") represents equations (59) plus the additional feedback term $-Bu$ and some external forcing input (or disturbance) w.
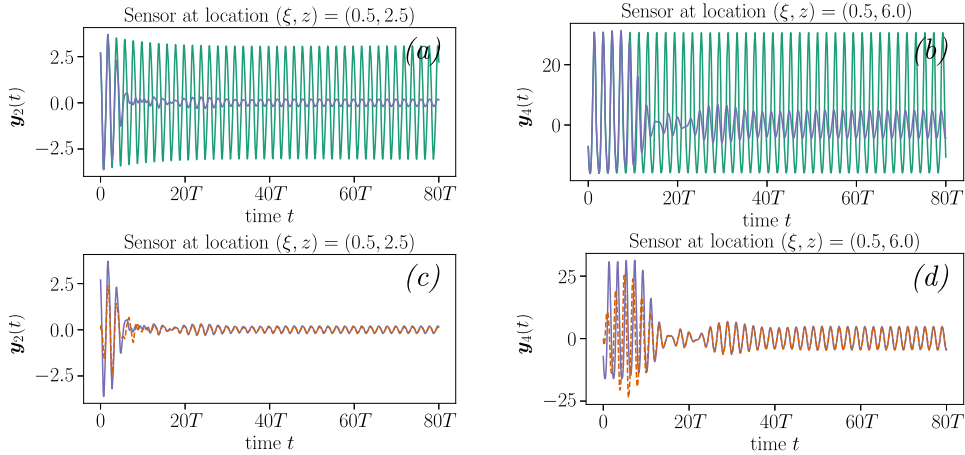

Fig. 4. $Re = 1250$. In panels *(a)* and *(b)* we show the sensor measurement from two of the four sensors from the simulation without control (green) and with feedback control (blue). In panels *(c)* and *(d)* we show the sensor measurement from the simulation with control (blue) and the predicted sensor measurement from the reduced-order estimator (orange).

7.4. Suppressing vortex pairing at $Re = 1250$

We begin by considering the case $Re = 1250$, where the T-periodic base flow is linearly stable. However, as previously mentioned, almost every small perturbation triggers vortex pairing. Here, we wish to design a reduced-order feedback controller and observer to suppress vortex pairing. We begin by computing a reduced-order model using the balancing procedure described in Algorithms 1 and 2. The balanced model has period $2T$ (recall that it is possible that the ROM has a higher period than the underlying full-order model) and we select model size $r = 6$. The truncation rank is chosen based on the decay of the singular values and on the predictive accuracy of the ROM, both shown in Appendix B. Using the reduced-order model, we design a feedback controller using the strategy discussed in section 7.3, and we select $Q_q(t)$ in (68) with

$$\alpha = \frac{10^{-3}}{\max_t \left(\mathbf{\Phi}(t)^* \boldsymbol{\xi}(t) \boldsymbol{\xi}(t)^* \mathbf{\Phi}(t) \right)}. \tag{73}$$

We also design an estimator and we choose $\beta = 10$. At this point we are ready to verify if we can suppress (or at least mitigate) the vortex pairing phenomenon. Since we know that vortex pairing is driven almost exclusively by the mode $\boldsymbol{\xi}(t)$ in Padovan and Rowley (2022), we induce vortex pairing by forcing the nonlinear Navier-Stokes equations (59) with the external forcing input

$$\boldsymbol{w}(t) = 10^{-3} \boldsymbol{\xi}(t), \quad \boldsymbol{w}(t) = \boldsymbol{w}(t + 2T), \tag{74}$$

where $\boldsymbol{\xi}(t)$ is normalized as in (67). The initial condition to the nonlinear full-order simulation is taken to be a state of heavy vortex pairing, while the initial condition for the reduced-order observer is set to zero. This is to emulate the realistic scenario where we do not have a good guess for the initial reduced-order state. We then integrate the (nonlinear) Navier-Stokes equations with and without feedback control, using the observer-based feedback configuration shown in Fig. 3. The results are shown in Figs. 4 and 6.

We begin with a brief discussion of Fig. 4, where we show the time history of the measurements from sensors 2 and 4 located at $(\xi, z) = (0.5, 2.5)$ and $(\xi, z) = (0.5, 6)$, respectively. In panels *(a)* and *(b)* we show the measured velocity of the flow with no control (green) and with observed-based feedback (blue). As hoped, we observe a significant reduction in the oscillation amplitude at both sensor locations when the feedback is active, meaning that the controller is successfully rejecting the disturbances that we are injecting into the flow. Panels *(c)* and *(d)* show the measured velocity of the flow with observer-based feedback (blue) and the predicted velocity from the reduced-order observer. Here, we see that at early times there are some discrepancies due to the fact that the nonlinear simulation is initialized with a non-zero initial condition, while the observer is initialized at zero. At later times, we see that the predicted output converges to the ground truth output, and this explains the success of the reduced-order controller/observer system in suppressing the oscillatory behavior of the flow.
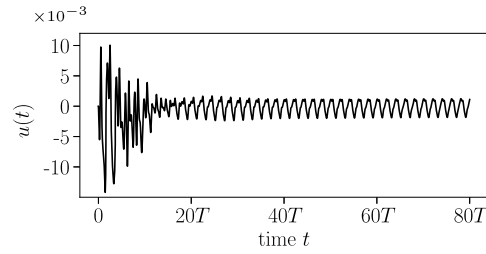
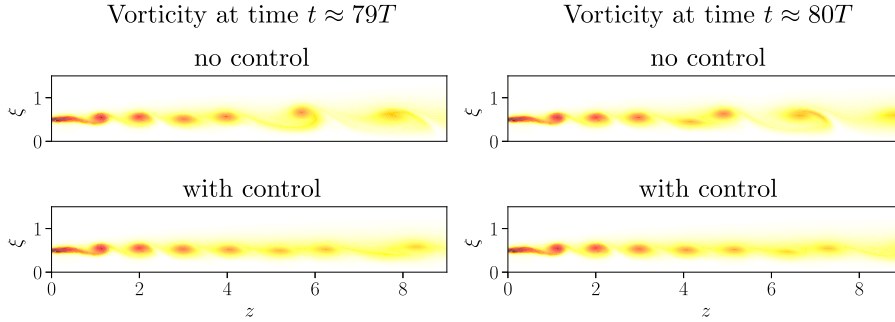**Fig. 5.** Control input $u(t)$ for the closed-loop simulation at $Re = 1250$.

Fig. 6. $Re = 1250$. Vorticity snapshots at times $t \approx 79T$ and $t \approx 80T$ from the response with and without feedback control. The colorbar is the same as in Fig. 2.

In Fig. 6 we show representative vorticity snapshots from the simulations with and without feedback control. Without control, the vortex rings pair at an axial location around $z \approx 5$. When the flow is controlled, we see that vortex pairing is significantly delayed/mitigated, and vortex rings pair further downstream, at around $z \approx 8$. We conclude by observing that this controller/observer pair was able to modify the flow dynamics with a control input that never exceeds $O(10^{-2})$ and actually remains below 5×10^{-3} for most of the times (see Fig. 5). This is indicative of the fact that the actuator is placed at a location where we have large control authority, and we therefore only require small-amplitude perturbations to modify the flow behavior. In a more practical setting, this means that this controller requires a low external energy supply, since it should not be energetically expensive to provide a velocity perturbation with magnitude equal to one thousandth of the flow characteristic velocity.

7.5. Suppressing vortex pairing at $Re = 1500$

We now consider $Re = 1500$, for which the T-periodic base flow is unstable, and the flow spirals onto a $2T$-periodic limit cycle characterized by pairing vortex rings. As before, we compute a ROM of dimension $r = 6$ (see Appendix C for details), and we design a controller as well as an observer. We choose the LQR weight $\boldsymbol{Q}_q(t)$ as in (68) with α given in (73). Similarly, the estimator weight is chosen as $\beta = 10$. As in the previous section, we externally force the flow with the "most dangerous" forcing profile (74), except that the magnitude is set to 10^{-4}. Since the base flow is unstable, vortex pairing will naturally occur even without the external forcing input. Here, however, we use this input to emulate the presence of external disturbances that perturb the flow on top of the underlying instability. The initial condition for the nonlinear full-order simulation is taken to be a state of heavy vortex pairing, while the initial condition for the reduced-order observer is set to zero. As explained in the previous section, this is a realistic choice based on the fact that we often lack knowledge of the initial state of the system. We then integrate the nonlinear Navier-Stokes equations in open loop and in closed-loop with the observed-based feedback configuration shown in Fig. 3. Results analogous to those shown in the previous section are shown in Fig. 7.

In this figure we see that the controller/observer pair is capable of suppressing the highly oscillatory behavior of the flow at all four sensor locations. Moreover, we see that the prediction of the measured output provided by the reduced-order estimator agrees well with the ground-truth measurements, especially at long times (see Figs. 7c and 7d). As in the previous section, the initial discrepancies arise because the observer was initialized with a zero initial condition, while the full-state had a non-zero initial condition.

Representative snapshots of the simulations with and without feedback control are shown in Fig. 9 at two different time instances. Remarkably, we see that while the uncontrolled flow exhibits strong vortex pairing (much stronger than at $Re = 1250$), the flow with feedback control does not. This means that the controller/observer pair successfully managed to suppress the underlying instability and to reject the most dangerous disturbance that we are injecting into the flow. Finally, we see from Fig. 8 that the feedback law generates a control input with magnitude $O(10^{-2})$ at early times, but once the upstream oscillations (see, e.g., Fig. 7a) have been suppressed, the required control input drops by an order of magnitude. We would like to point out that even at early times, the demanded control input is significantly lower in magnitude than the flow characteristic velocity.
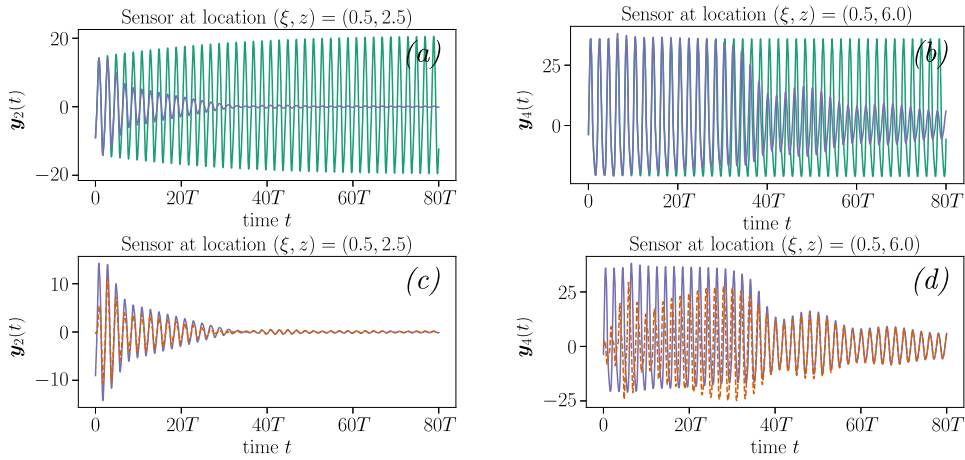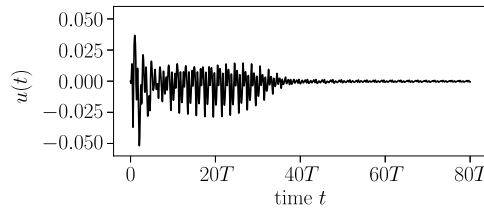
Fig. 7. Analog of Fig. 4 for $Re = 1500$.

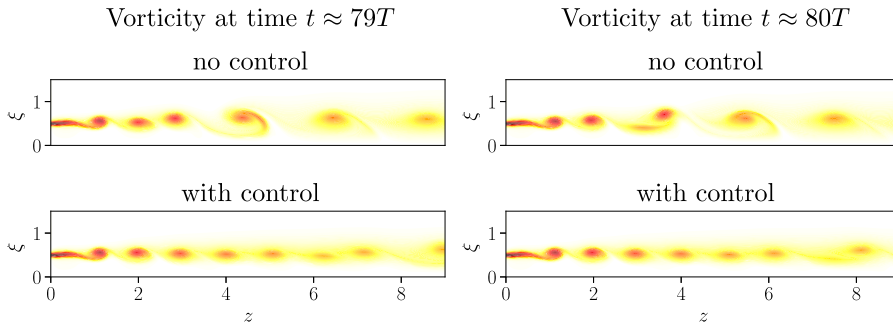**Fig. 8.** Control input $u(t)$ for the closed-loop simulation at $Re = 1500$.

Fig. 9. Analog of Fig. 6 at $Re = 1500$.

8. Conclusion

In this paper we perform continuous-time balanced truncation for time-periodic systems using the frequency-domain representation of the reachability and observability Gramians. We have seen that these frequential Gramians are well-defined both when the underlying system is stable and when it is unstable. Moreover, when have seen that computing the Gramians using their frequency-domain representation can offer computational benefits, especially if the dynamics exhibit slowly-decaying transients. We demonstrated this approach on a periodically-forced axisymmetric jet at Reynolds numbers $Re = 1250$ and $Re = 1500$ (corresponding to stable and unstable equilibria), and in both cases we used the balanced model to design reduced-order controllers and observers to suppress the vortex pairing phenomenon.

CRediT authorship contribution statement

Alberto Padovan: Conceptualization, Methodology, Software, Writing – original draft. **Clarence W. Rowley:** Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

Appendix A. Proofs

A.1. Proof of Proposition 2

We prove the proposition for $\boldsymbol{P}(t)$, since the result for $\boldsymbol{Q}(t)$ follows analogously. Let us define the quantity

$$\hat{\boldsymbol{P}}_k = \sum_{\substack{m,l \in \mathbb{Z} \\ m=k+l}} \hat{\boldsymbol{P}}_{m-l} := \sum_{\substack{m,l \in \mathbb{Z} \\ m=k+l}} \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{Z}_m(\gamma) \boldsymbol{Z}_l(\gamma)^* \mathrm{d}\gamma. \tag{A.1}$$

Given the definition of $\boldsymbol{Z}_m(\gamma)$ in (18) and using Proposition 1, it follows that $\hat{\boldsymbol{P}}_{m-l}$ satisfies the algebraic Sylvester equation below

$$(-im\omega \boldsymbol{I} + \boldsymbol{J}) \hat{\boldsymbol{P}}_{m-l} + \hat{\boldsymbol{P}}_{m-l} \left(il\omega \boldsymbol{I} + \boldsymbol{J}^* \right) + \mathcal{P}_s \widetilde{\boldsymbol{B}}_m \widetilde{\boldsymbol{B}}_l^* \mathcal{P}_s - \mathcal{P}_u \widetilde{\boldsymbol{B}}_m \widetilde{\boldsymbol{B}}_l^* \mathcal{P}_u = 0. \tag{A.2}$$

Summing over m and l as in (A.1), writing $m = k + l$, and using the linearity of the Sylvester equation, one can see that $\hat{\boldsymbol{P}}_k$ satisfies (13). In other words, $\hat{\boldsymbol{P}}_k$ is the kth Fourier coefficient of $\boldsymbol{P}(t)$. Writing $\boldsymbol{P}(t)$ as

$$\boldsymbol{P}(t) = \sum_{k \in \mathbb{Z}} \hat{\boldsymbol{P}}_k e^{ik\omega t} = \sum_{k \in \mathbb{Z}} \sum_{\substack{m,l \in \mathbb{Z} \\ m=k+l}} \hat{\boldsymbol{P}}_{m-l} e^{i(m-l)\omega t} = \sum_{m,l \in \mathbb{Z}} \hat{\boldsymbol{P}}_{m-l} e^{i(m-l)\omega t} \tag{A.3}$$

concludes the proof.

A.2. Proof of Proposition 3

We need to show that

$$\boldsymbol{H}_{k,j}(\gamma) = \sum_{m \in \mathbb{Z}} \boldsymbol{V}_{k-m} (i\gamma \boldsymbol{I} - (-im\omega \boldsymbol{I} + \boldsymbol{J}))^{-1} \boldsymbol{W}_{j-m}^*. \tag{A.4}$$

Let us start from (2) and write $\boldsymbol{z}(t)$ and $\boldsymbol{u}(t)$ as EMP signals (as in (26)), to obtain

$$\boldsymbol{z}_{m+\gamma} = (i\gamma \boldsymbol{I} - (-im\omega \boldsymbol{I} + \boldsymbol{J}))^{-1} \sum_{j,l \in \mathbb{Z}} \boldsymbol{W}_{j-m}^* \boldsymbol{B}_{j-l} \boldsymbol{u}_{l+\gamma}. \tag{A.5}$$

Using $\boldsymbol{x}(t) = \boldsymbol{V}(t)\boldsymbol{z}(t)$, the coefficient $\boldsymbol{x}_{k+\gamma}$ is given by

$$\sum_{m \in \mathbb{Z}} \boldsymbol{V}_{k-m} \boldsymbol{z}_{m+\gamma} = \sum_{m,j,l \in \mathbb{Z}} \boldsymbol{V}_{k-m} (i\gamma \boldsymbol{I} - (-im\omega \boldsymbol{I} + \boldsymbol{J}))^{-1} \boldsymbol{W}_{j-m}^* \boldsymbol{B}_{j-l} \boldsymbol{u}_{l+\gamma}. \tag{A.6}$$

Comparing (29) and (A.6) shows that (A.4) indeed holds, and this concludes the proof.

A.3. Proof of Proposition 4

The existence of such integer m is immediate. Let us consider the quantity $\boldsymbol{Z}_{\boldsymbol{x},k+\alpha} e^{ik\omega t} := \sum_{j \in \mathbb{Z}} \boldsymbol{H}_{k,j}(\alpha) \boldsymbol{B}_j e^{ik\omega t}$, which, from the definition of $\boldsymbol{H}$, satisfies

$$i(\alpha + k\omega) \boldsymbol{Z}_{\boldsymbol{x},k+\alpha} e^{ik\omega t} = \sum_{l \in \mathbb{Z}} \boldsymbol{A}_{k-l} \boldsymbol{Z}_{\boldsymbol{x},l+\alpha} e^{ik\omega t} + \boldsymbol{B}_k e^{ik\omega t}. \tag{A.7}$$

Substituting $\alpha = \gamma + m\omega$ and manipulating the indices inside the sum, we obtain

$$i(\gamma + (k+m)\omega) \boldsymbol{Z}_{\boldsymbol{x},(k+m)+\gamma} e^{ik\omega t} = \sum_{l \in \mathbb{Z}} \boldsymbol{A}_{(k+m)-l} \boldsymbol{Z}_{\boldsymbol{x},l+\gamma}^{(j)} e^{ik\omega t} + \boldsymbol{B}_k e^{ik\omega t}. \tag{A.8}$$

Changing variables according to $n = k + m$, we have
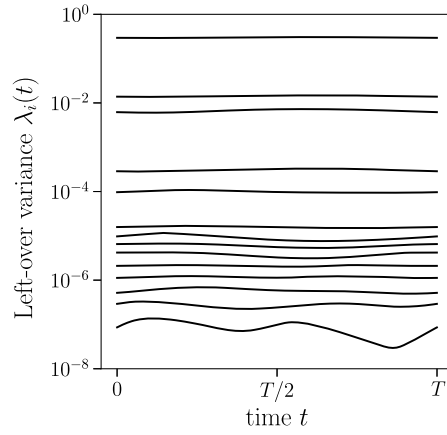
Fig. B.10. Left-over variance (B.1), computed from the Hankel singular values of $Y(t)^*X(t)$ at $Re = 1250$.
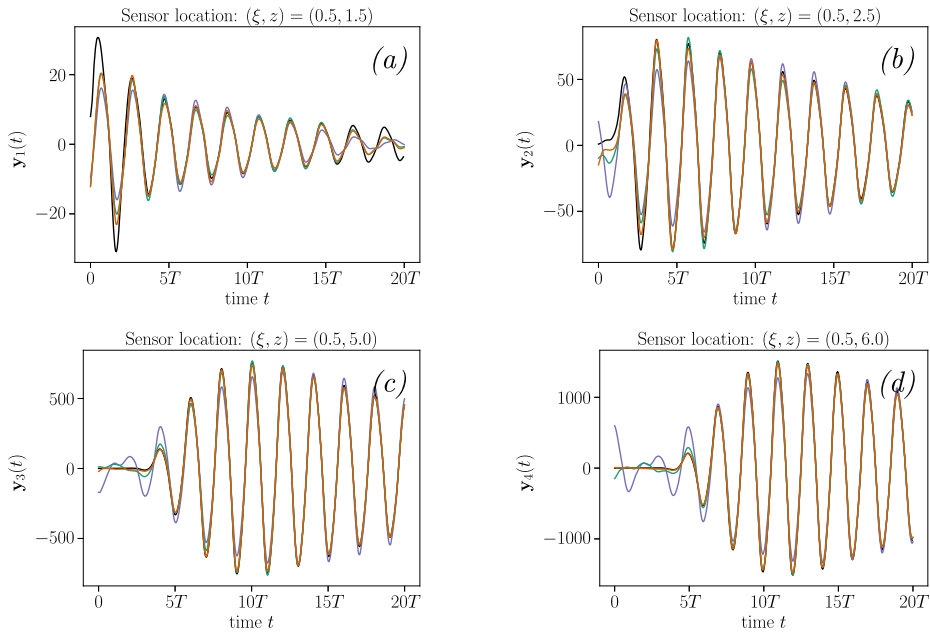

Fig. B.11. Measured output $y_i(t)$ at all four sensor location from the linear impulse response at $Re = 1250$. Ground truth (black), ROM with size $r = 2$ (purple), ROM with size $r = 4$ (green), ROM with size $r = 6$ (orange).

$$i(\gamma + n\omega)Z_{x,n+\gamma}e^{i(n-m)\omega t} = \sum_{l \in \mathbb{Z}} A_{n-l}Z_{x,l+\gamma}e^{i(n-m)\omega t} + B_{n-m}e^{i(n-m)\omega t}. \tag{A.9}$$

Thus $Z_{x,k+\alpha}e^{ik\omega t} = \sum_{j \in \mathbb{Z}} H_{k,j}(\gamma)B_{j-m}e^{i(k-m)\omega t}$, and this concludes the proof.

Appendix B. Reduced-order models at $Re = 1250$

In this section we study the performance of different reduced-order models (ROMs) as a function of the model size. Throughout, the Reynolds number $Re = 1250$, which gives us a linearly-stable periodic base flow with period T.

We begin by computing the Hankel singular values $\sigma_i(t)$ of the product $Y(t)^*X(t)$ (see Algorithm 2). From these singular values, we can compute the left-over variance, defined as

$$\lambda_i(t) = 1 - \frac{\sum_{j=1}^{i} \sigma_j(t)^2}{\sum_{j=1}^{N} \sigma_j(t)^2}. \tag{B.1}$$

This quantity is shown in Fig. B.10, and we see that the input-output dynamics of the jet flow at $Re = 1250$ are very low rank since the first few Hankel singular values capture the greatest majority of the variance.
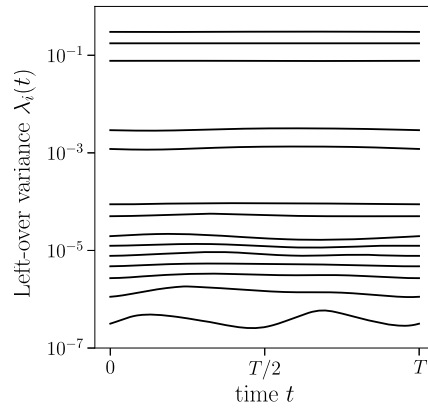
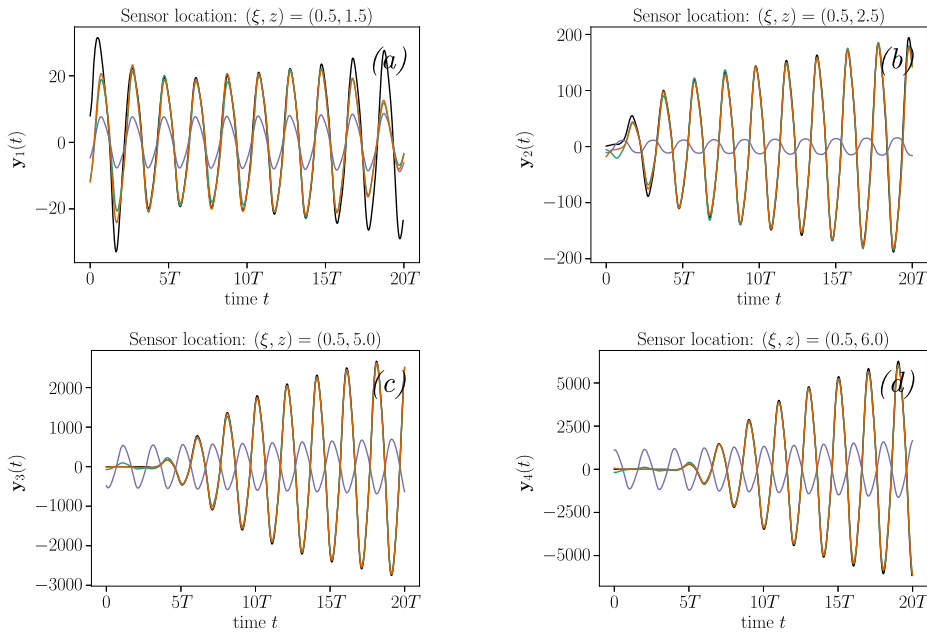**Fig. C.12.** Analog of Fig. B.10 at $Re = 1500$.

Fig. C.13. Analog of Fig. B.11 at $Re = 1500$.

Moving forward, we select ROM sizes $r = 2$, $r = 4$ and $r = 6$, and we study the performance of these ROMs. In particular, we wish to see how well they can predict the output $y(t)$ in response to (linear) impulses $Bu(t) = B\delta(t - \tau)$. In other words, we compare the ROM to the ground truth obtained from numerical integration of the linearized Navier-Stokes equations. The outputs from an impulse at time $\tau = 0$ are shown in Fig. B.11. Here we see that as we increase the ROM dimension (i.e., as we capture more of the variance), the predictive capabilities of the ROM improve. In particular, even at $r = 4$, the ROM is capable of correctly predicting the amplitude and phase of the response. With $r = 6$, we further improve on the early-time prediction. For completeness, it is worth mentioning that the ROMs have similar performance also for impulses at times $\tau \neq 0$ (although we do not show the corresponding plots here).

Appendix C. Reduced-order models at $Re = 1500$

Here, we perform the same analysis as in Appendix B, except that we consider $Re = 1500$. Recall that for this configuration the base flow is linearly unstable. The left-over variance is shown in Fig. C.12, while the measured output from a linear impulse response at time $\tau = 0$ is shown in Fig. C.13. In the latter, we see that the ROMs of sizes $r = 4$ and $r = 6$ correctly predict the amplitude and phase of the measured outputs, as well as the linear growth rate due to the instability in the underlying base flow.

References

[1] S. Ahuja, C.W. Rowley, Feedback control of unstable steady states of flow past a flat plate using reduced-order estimators, J. Fluid Mech. 645 (2010) 447–478.

[2] P.R. Amestoy, I.S. Duff, J. Koster, J.-Y. L'Excellent, A fully asynchronous multifrontal solver using distributed dynamic scheduling, SIAM J. Matrix Anal. Appl. 23 (2001) 15–41.
[3] S. Bittanti, P. Colaneri, G. Guardabassi, Periodic solutions of periodic Riccati equations, IEEE Trans. Autom. Control 29 (1984) 665–667.
[4] P. Bolzern, P. Colaneri, The periodic Lyapunov equation, SIAM J. Matrix Anal. Appl. 9 (1988) 499–512, https://doi.org/10.1137/0609041.
[5] J.L. Chern, L. Dieci, Smoothness and periodicity of some matrix decompositions, SIAM J. Matrix Anal. Appl. 22 (2001) 772–792, https://doi.org/10.1137/S0895479899353622.
[6] G. Dergham, D. Sipp, J.C. Robinet, A. Barbagallo, Model reduction for fluids using frequential snapshots, Phys. Fluids 23 (2011) 064101.
[7] G.E. Dullerud, F.G. Paganini, A Course in Robust Control Theory: A Convex Approach, Springer, New York, 2000.
[8] M. Farhood, C.L. Beck, G.E. Dullerud, Model reduction of periodic systems: a lifting approach, Automatica 41 (2005) 1085–1090.
[9] T.L.B. Flinois, A.S. Morgans, P.J. Schmid, Projection-free approximate balanced truncation of large unstable systems, Phys. Rev. E 92 (2015) 023012.
[10] G. Floquet, Sur les équations différentielles linéaires à coefficients périodiques, Ann. Sci. Éc. Norm. Supér. (2) 12 (1883) 47–88.
[11] S.K. Godunov, Modern Aspects of Linear Algebra, American Mathematical Society, 1998.
[12] W. Johnson, Helicopter Theory, Princeton University Press, 1980.
[13] M.R. Jovanović, From bypass transition to flow control and data-driven turbulence modeling: an input–output viewpoint, Annu. Rev. Fluid Mech. 53 (2021) 311–345.
[14] M.R. Jovanović, M. Fardad, H_2 norm of linear time-periodic systems: a perturbation analysis, Automatica 44 (2008) 2090–2098.
[15] O. Koch, C. Lubich, Dynamical low-rank approximation, SIAM J. Matrix Anal. Appl. 29 (2007) 434–454.
[16] N. Lang, J. Saak, T. Stykel, Balanced truncation model reduction for linear time-varying systems, Math. Comput. Model. Dyn. Syst. 22 (2016) 267–281.
[17] S. Longhi, G. Orlando, Balanced reduction of linear periodic systems, Kybernetika 35 (1999) [737]–751.
[18] C. Lubich, I.V. Oseledets, A projector-splitting integrator for dynamical low-rank approximation, BIT Numer. Math. 54 (2014) 171–188.
[19] Z. Ma, Reduction and reconstruction methods for simulation and control of fluids, Ph.D. thesis, Princeton University, 2010.
[20] Z. Ma, C.W. Rowley, G. Tadmor, Snapshot-based balanced truncation for linear time-periodic systems, IEEE Trans. Autom. Control 55 (2010) 469–473.
[21] R. Moarref, M.R. Jovanović, Controlling the onset of turbulence by streamwise travelling waves. Part 1. Receptivity analysis, J. Fluid Mech. 663 (2010) 70–99, https://doi.org/10.1017/S0022112010003393.
[22] R. Moarref, M.R. Jovanović, Model-based design of transverse wall oscillations for turbulent drag reduction, J. Fluid Mech. 707 (2012) 205–240, https://doi.org/10.1017/jfm.2012.272.
[23] B. Moore, Principal component analysis in linear systems: controllability, observability, and model reduction, IEEE Trans. Autom. Control 26 (1981) 17–32.
[24] A. Padovan, S.E. Otto, C.W. Rowley, Analysis of amplification mechanisms and cross-frequency interactions in nonlinear flows via the harmonic resolvent, J. Fluid Mech. 900 (2020) A14, https://doi.org/10.1017/jfm.2020.497.
[25] A. Padovan, C.W. Rowley, Analysis of the dynamics of subharmonic flow structures via the harmonic resolvent: application to vortex pairing in an axisymmetric jet, Phys. Rev. Fluids 7 (2022) 073903, https://doi.org/10.1103/PhysRevFluids.7.073903.
[26] W. Ran, A. Zare, M.R. Jovanović, Model-based design of riblets for turbulent drag reduction, J. Fluid Mech. 906 (2021) A7.
[27] C.W. Rowley, Model reduction for fluids, using balanced proper orthogonal decomposition, Int. J. Bifurc. Chaos 15 (2005) 997–1013.
[28] H. Sandberg, A. Rantzer, Balanced truncation of linear time-varying systems, IEEE Trans. Autom. Control 49 (2004) 217–229.
[29] L. Shaabani-Ardali, D. Sipp, L. Lesshafft, Time-delayed feedback technique for suppressing instabilities in time-periodic flow, Phys. Rev. Fluids 2 (2017) 113904.
[30] L. Shaabani-Ardali, D. Sipp, L. Lesshafft, Vortex pairing in jets as a global Floquet instability: modal and transient dynamics, J. Fluid Mech. 862 (2019) 951–989.
[31] A. Varga, Balanced truncation model reduction of periodic systems, in: Proceedings of the 39th IEEE Conference on Decision and Control, 2000, pp. 2379–2384.
[32] N.M. Wereley, Analysis and control of linear periodically time varying systems, Ph.D. thesis, Massachusetts Institute of Technology, 1991.
[33] W. Wu, C. Meneveau, R. Mittal, A. Padovan, C.W. Rowley, L. Cattafesta, Response of a turbulent separation bubble to zero-net-mass-flux jet perturbations, Phys. Rev. Fluids 7 (2022) 084601, https://doi.org/10.1103/PhysRevFluids.7.084601.