

# PERIODIC EIGENDECOMPOSITION AND ITS APPLICATION TO KURAMOTO-SIVASHINSKY SYSTEM \*

XIONG DING <sup>†</sup> AND PREDRAG CVITANOVIĆ <sup>†</sup>

**Abstract.** Periodic eigendecomposition algorithm for calculating eigenvectors of a periodic product of a sequence of matrices, an extension of the periodic Schur decomposition, is formulated and compared with the recently proposed covariant vectors algorithms. In contrast to those, periodic eigendecomposition requires no power iteration and is capable of determining not only the real eigenvectors, but also the complex eigenvector pairs. Its effectiveness, and in particular its ability to resolve eigenvalues whose magnitude differs by hundreds of orders, is demonstrated by applying the algorithm to computation of the full linear stability spectrum of periodic solutions of Kuramoto-Sivashinsky system.

**Key words.** periodic eigendecomposition, periodic Schur decomposition, periodic Sylvester equation, Francis algorithm, simultaneous iteration, covariant Lyapunov vectors, Floquet vectors, Kuramoto-Sivashinsky, linear stability, stability multipliers, Lyapunov exponents, Cauchy-Green strain tensor

**AMS subject classifications.** 15A18, 37M25, 65F15, 65P20, 65P40

**1. Introduction.** In this paper we implement the periodic Schur decomposition algorithm for computation of eigenspectra and eigenvectors of periodic products of matrices [2, 17], adopt it to the problem of determining the linear stability of periodic solutions of high-dimensional nonlinear dynamical systems, and compare it to the recently introduced algorithms [31, 15, 14] for calculating the linear stability spectra and the associated ‘covariant vectors’ for ergodic, hyperbolically unstable trajectories of such flows. A critical re-examination of these algorithms is necessitated by the fact that in nonlinear dynamics the matrices typically act on  $10$  to  $10^6$ -dimensional vector spaces, the matrix elements can easily vary over  $100$ ’s or  $1000$ ’s orders of magnitude, and yet in the applications one sometimes needs to determine their eigenspectra and eigenvectors to a very high accuracy.

*Covariant vectors algorithm* [14] is a combination of forward iteration and backward iteration that relies on the convergence to the Gram-Schmidt vectors in the transient evolution [9]. When applied to unstable periodic solutions of a nonlinear dynamical system, it is implemented as a combination of simultaneous iteration and pure power iteration such that covariant vectors converge to Floquet vectors or subspaces spanned by complex Floquet vector pairs. Applied to periodic orbits, the algorithm has several drawbacks. First, it assumes that all eigenvalues have distinct magnitudes, but a periodic orbit can have complex eigenvalue pairs of the same magnitude. They cannot be separated by power iteration, and thus require special attention. Second, the convergence rate of the backward iteration is linearly dependent on the ratio of magnitudes of eigenvalues, so the algorithm is not efficient if eigenvalues cluster.

An algorithm for computation of eigenvalues of the *periodic Schur decomposition* of a product of matrices was given by Bojanczyk *et al.* [2], as an extension to the standard QR iteration (Francis algorithm [12]). The eigenvectors can then be obtained by Granat *et al.* [17] reordering algorithm for a given periodic real Schur form (PRSF). The algorithm can switch any two diagonal blocks and relies on the periodic QR algorithm to restore PRSF. For our purposes, it suffices to consider two special cases and

---

\*This work was supported by the NSF grant 1028133, and an endowment from the family of Glen Robinson, Jr..

<sup>†</sup>Center for Nonlinear Science, School of Physics, Georgia Institute of Technology, Atlanta

the relation between the solution of periodic Sylvester equation and its eigenvectors in order to obtain periodic eigendecomposition.

Here we implement and compare these two different approaches to calculating Floquet vectors: the covariant vectors algorithm and the periodic eigendecomposition algorithm. There are two stages in the process of calculating Floquet vectors, each of which can be accomplished by two different methods, so we study performance of four different algorithms in all. The paper is organized as follows. Sect. 2 describes briefly the nonlinear dynamics motivation for undertaking this project, and can be skipped by a reader interested only in the algorithms. We describe the computational problem in sect. 3. In sect. 4 we deal with the first stage of periodic eigendecomposition, and then show that both the periodic QR algorithm and simultaneous iteration are capable of achieving periodic Schur decomposition. Sect. 5 introduces power iteration and reordering as two practical methods to obtain all eigenvectors. In sect. 6 we compare the computational effort required by different methods, and sect. 7 applies periodic eigendecomposition to Kuramoto-Sivashinsky equation, an example which illustrates method's effectiveness.

**2. Dynamics and periodic eigendecomposition.** Partial differential equations, such as Navier-Stokes equations of fluid dynamics, are in principle  $\infty$ -dimensional dynamical systems. However, recent work of Yang *et al.* offers strong numerical evidence [32] that the chaotic solutions of two spatially extended systems, Kuramoto-Sivashinsky and complex Landau-Ginzburg, evolve within a manifold spanned by a *finite number* of ‘entangled’ modes, dynamically isolated from the residual set of isolated, transient degrees of freedom, in agreement with the rigorous bounds on dimensions of inertial manifolds for dissipative PDEs [3]. This work is motivated by recent algorithms for computation of large numbers of ‘covariant vectors’, the eigenvectors of the linearization of flow around ergodic trajectories [23, 31, 22, 15, 32, 24, 14]. Covariant vectors exhibit an approximate orthogonality between the ‘entangled’ modes and the rest, the ‘isolated’ modes. These results suggest that for a faithful numerical integration of dissipative PDEs, a finite number of entangled modes should suffice, and that increasing the dimensionality beyond that merely increases the number of isolated modes, with no effect on the long-time dynamics.

While these studies offer strong evidence for finite dimensionality of chaotic (or ‘turbulent’) attractors of dissipative flows, they are based on numerical simulations of long ergodic trajectories and they yield no intuition about the geometry of the attractor. That is attained by studying the hierarchies of unstable periodic orbits, invariant solutions which, together with their Floquet vectors, provide an effective description of both the local hyperbolicity and the global geometry of an attractor embedded in a high-dimensional state space. Motivated by the above studies of covariant vectors, we formulate in this paper a periodic eigendecomposition algorithm suited to accurate computation of Floquet vectors of unstable periodic orbits. We start by defining the relevant nonlinear dynamics concepts, following conventions of ChaosBook.org [5].

Let the flow of a continuous time system be described by the time-forward map  $x(t) = f^t(x_0)$ ,  $x \in \mathbb{R}^n$ . In the linear approximation, the deformation of an infinitesimal neighborhood of  $x(t)$  (dynamics in tangent space) is governed by the Jacobian matrix  $\delta x(x_0, t) = J^t(x_0) \delta x(x_0, 0)$ , where  $J^t(x_0) = J^{t-t_0}(x_0, t_0) = \partial f^t(x_0) / \partial x_0$ . For a periodic point  $x$  on orbit  $p$  of period  $T_p$ ,  $J_p = J^{T_p}(x)$  is called the Floquet matrix and its eigenvalues the Floquet multipliers  $\Lambda_j$ . The associated Floquet eigenvectors  $\mathbf{e}_j(x)$ ,  $J_p \mathbf{e}_j = \Lambda_j \mathbf{e}_j$ , define the invariant directions of the tangent space at the periodic point  $x = x(t) \in p$ . Floquet multipliers are either real,  $\Lambda_j = \sigma_j |\Lambda_j|$ ,  $\sigma_j \in \{1, -1\}$ , or

form complex pairs,  $\{\Lambda_j, \Lambda_{j+1}\} = \{\exp(i\theta_j)|\Lambda_j|, \exp(-i\theta_j)|\Lambda_j|\}$ ,  $0 < \theta_j < \pi$ . Floquet exponents  $\mu_j = (\ln |\Lambda_j|)/T_p$  describe the mean contraction or expansion rates per one period of the orbit.

A Floquet multiplier is a dimensionless ratio of the final/initial perturbation along the  $j_{th}$  eigen-direction. It is an intrinsic, local property of a smooth flow, invariant under any smooth coordinate transformation. In case that the state space is equipped with a notion of distance, one can, following Lyapunov [20], characterize the mean growth rate of the distance between neighboring trajectories during time  $t$ , by the leading Lyapunov exponent  $\lambda \simeq \ln(\|\delta x(t)\|/\|\delta x(0)\|)/t$ . More precisely, if the norm is Euclidean, one defines the finite-time Lyapunov or characteristic exponents,

$$(2.1) \quad \lambda(x_0, \hat{n}; t) = \frac{1}{t} \ln \|J^t \hat{n}\| = \frac{1}{2t} \ln (\hat{n}^\top J^t J^t \hat{n}),$$

where  $J^\top J$  is the right Cauchy-Green strain tensor of continuum mechanics. If the unit vector  $\hat{n}$ ,  $\|\hat{n}\| = 1$  is aligned along the  $j_{th}$  principal stretch at the initial time,  $\hat{n} = u_j$ , then the corresponding finite-time Lyapunov exponent is given by

$$(2.2) \quad \lambda_j(x_0; t) = \lambda(x_0, u_j; t) = \frac{1}{t} \ln \sigma_j(x_0; t),$$

where  $\sigma_j$  is the  $j_{th}$  singular value of matrix  $J^t(x_0)$ .

So there are two ways of characterizing the (in)stability of a dynamical system. Eigenvectors / eigenvalues are suited to study of iterated forms of a matrix, such as Jacobian matrix  $J$ , and are thus a natural tool for study of dynamics. Singular values of the strain tensor  $J^\top J$  are suited to study of the matrix  $J$  itself, and are often used because the singular value decomposition is convenient for numerical work: there is vast literature on numerical computation of Lyapunov exponents, see for example refs. [30, 8, 7, 25]. Singular values  $\{\sigma_j\}$  are not related to the Floquet multipliers  $\{\Lambda_j\}$  in any simple way [28]. Floquet multipliers are invariant under all local smooth nonlinear coordinate transformations, they are intrinsic to the flow, and Floquet eigenvectors depend on no norm. Covariant vectors / Floquet eigenvectors map forward and backward as  $\mathbf{e}_j \rightarrow J \mathbf{e}_j$  under time evolution and remain tangent to the attractor. In contrast, the principal axes point away from it and are not covariant, i.e., the linearized dynamics does not transport them into the tangent space computed further downstream. Furthermore, the principal axes have to be recomputed from the scratch for each time  $t$  since the strain tensor  $J^\top J$  satisfies no multiplicative group property (2.3): unlike the periodic orbit Jacobian matrix, the strain tensor  $(J^\top)^r J^r$  for the  $r_{th}$  repeat of a prime cycle is not given by a power of  $J^\top J$  for the single traversal of the prime cycle. The deep reason for these shortcomings of singular values  $\{\sigma_j\}$  and principal axes is that they depend on the choice of a norm (2.1). A norm is largely arbitrary, and externally imposed upon the dynamics. The Euclidean (or  $L^2$ ) distance is natural in the theory of 3D continuous media, but what the norm should be for other state spaces is far from clear, especially in high dimensions and for discretizations of PDEs.

The connection between the two characterizations is asymptotic in time, and provided by the Oseledec Multiplicative Ergodic Theorem [21] which states that the long time limits of (2.2) exist for almost all points  $x_0$  and vectors  $\hat{n}$ , and that there are at most  $n$  distinct Lyapunov exponents  $\lambda_j(x_0)$  as  $\hat{n}$  ranges over the tangent space. For periodic orbits these  $\lambda_j$  (evaluated numerically as  $t \rightarrow \infty$  limits of many repeats of the prime period  $T$ ) coincide with Floquet exponents  $\mu_j$  (computed in one period of the orbit).

To summarize, based on geometric considerations, we would much prefer to compute Floquet eigenspectrum of a periodic orbit  $J$  directly, rather than via the singular values,  $J^\top J$  detour. Unfortunately, the Floquet matrix and its spectrum cannot be easily computed, as the magnitude of matrix elements may range over 100's or more orders of magnitude. However –and demonstrating this is the main goal of this paper– the group property of Jacobian matrix multiplication (chain rule) along the orbit,

$$(2.3) \quad J^{t-t_0}(x(t_0), t_0) = J^{t-t_1}(x(t_1), t_1) J^{t_1-t_0}(x(t_0), t_0),$$

enables us to factorize it into a product of short-time matrices with matrix elements of comparable magnitudes. Periodic eigendecomposition can then be used to calculate all Floquet multipliers and Floquet vectors along a periodic orbit.

**3. Description of the problem.** A product of  $m$  real matrices

$$(3.1) \quad \mathbf{J}^{(0)} = J_m J_{m-1} \cdots J_1, \quad J_i \in \mathbb{R}^{n \times n}, \quad i=1, 2, \dots, m$$

can be diagonalized if and only if the sum of dimensions of eigenspaces of  $\mathbf{J}^{(0)}$  is  $n$ .

$$(3.2) \quad \mathbf{J}^{(0)} = V^{(0)} D (V^{(0)})^{-1},$$

where  $D$  is a diagonal matrix which stores  $\mathbf{J}^{(0)}$ 's eigenvalues,  $\{\Lambda_1, \Lambda_2, \dots, \Lambda_n\}$ , and columns of matrix  $V^{(0)}$  are the eigenvectors of  $\mathbf{J}^{(0)}$ :  $V^{(0)} = [v_1^{(0)}, v_2^{(0)}, \dots, v_n^{(0)}]$ . In this paper all vectors are written in the column form, transpose of  $v$  is denoted  $v^\top$ , Euclidean ‘dot’ product by  $(v^\top u)$ , and bold capital letter represents a product of a sequence of matrices. The challenge associated with obtaining diagonalized form (3.2) is the fact that often  $\mathbf{J}^{(0)}$  cannot be written explicitly as the elements of  $\mathbf{J}^{(0)}$  can easily overflow or underflow numerically for large  $m$ . In nonlinear dynamics applications such as periodic orbit theory [5], each periodic orbit comes equipped with a set of Floquet multipliers  $\Lambda_j$  (eigenvalues of its Floquet matrix  $\mathbf{J}^{(0)}$ ) and Floquet vectors  $\mathbf{e}_j$  (eigenvectors of Floquet matrix). Floquet multipliers can easily vary over 100's orders of magnitude, depending on the system under study and the period of the orbit; therefore an algorithm of high accuracy is needed, if one is to resolve all these multipliers and eigenvectors. Also, not only the eigendecomposition of  $\mathbf{J}^{(0)}$  is required, but also the eigendecomposition of its cyclic rotations:  $\mathbf{J}^{(k)} = J_k J_{k-1} \cdots J_1 J_m \cdots J_{k+1}$  for  $k = 1, 2, \dots, m-1$ . Eigendecomposition of all  $\mathbf{J}^{(k)}$  is called the *periodic eigendecomposition* of the matrix sequence  $J_m, J_{m-1}, \dots, J_1$ .

The process of implementing eigendecomposition (3.2) proceeds in two stages. First, periodic real Schur form (PRSF) is obtained by a similarity transformation for each  $J_i$ ,

$$(3.3) \quad J_i = Q_i R_i Q_{i-1}^\top,$$

with  $Q_i$  orthogonal matrix, and  $Q_0 = Q_m$ . In the case considered here,  $R_m$  is quasi-upper triangular with  $[1 \times 1]$  and  $[2 \times 2]$  blocks on the diagonal, and the remaining  $R_i, i = 1, 2, \dots, m-1$  are upper triangular. The existence of PRSF, proved in ref. [2], provides the periodic QR algorithm that implements periodic Schur decomposition. Defining  $\mathbf{R}^{(k)} = R_k R_{k-1} \cdots R_1 R_m \cdots R_{k+1}$ , we have

$$(3.4) \quad \mathbf{J}^{(k)} = Q_k \mathbf{R}^{(k)} Q_k^\top,$$

with the eigenvectors of matrix  $\mathbf{J}^{(k)}$  related to eigenvectors of quasi-upper triangular matrix  $\mathbf{R}^{(k)}$  by orthogonal matrix  $Q_k$ .  $\mathbf{J}^{(k)}$  and  $\mathbf{R}^{(k)}$  have the same eigenvalues,

stored in the  $[1 \times 1]$  and  $[2 \times 2]$  blocks on the diagonal of  $\mathbf{R}^{(k)}$ , and their eigenvectors are transformed by  $Q_k$ , so the second stage concerns the eigendecomposition of  $\mathbf{R}^{(k)}$ . Eigenvector matrix of  $\mathbf{R}^{(k)}$  has the same structure as  $R_m$ . We evaluate it by two distinct algorithms. The first one is a combination of power iteration and shifted power iteration, while the second algorithm relies on solving a periodic Sylvester equation [17].

As all  $\mathbf{R}^{(k)}$  have the same eigenvalues, and their eigenvectors are related by similarity transformations,

$$(3.5) \quad \mathbf{R}^{(k)} = (R_m \cdots R_{k+1})^{-1} \mathbf{R}^{(0)} (R_m \cdots R_{k+1}),$$

one may be tempted to calculate the eigenvectors of  $\mathbf{R}^{(0)}$ , and obtain the eigenvectors of  $\mathbf{R}^{(k)}$  by (3.5). The pitfall of this approach is that numerical errors accumulate when multiplying a sequence of upper triangular matrices, especially for large  $k$ . Therefore, in the second stage of implementing periodic eigendecomposition, iteration is needed for each  $\mathbf{R}^{(k)}$  if power iteration algorithm is chosen in this stage. Periodic Sylvester equation bypasses this problem by giving the eigenvectors of all  $\mathbf{R}^{(k)}$ .

Our work illustrates the connection between different algorithms in the two stages of implementing periodic eigendecomposition, pays attention to the case when eigenvectors appear as complex pairs, and demonstrates that eigenvectors can be obtained directly from periodic Sylvester equation without restoring PRSF.

**4. Periodic Schur decomposition.** Eq. (3.4) represents the eigenvalues of matrix  $\mathbf{J}^{(k)}$  as real eigenvalues on the diagonal, and complex eigenvalue pairs as  $[2 \times 2]$  blocks on the diagonal of  $\mathbf{R}^{(k)}$ . Previous work [2, 19] implemented periodic QR algorithm to achieve the PRSF. We also implement simultaneous iteration for a sequence of matrices  $J_k$  to achieve the same goal. The two algorithms are equivalent [27], but their computational costs differ.

**4.1. Periodic QR algorithm.** Periodic Schur decomposition proceeds in two stages. First, matrices  $J_i$ ,  $i = 1, 2, \dots, m$  are transformed to upper Hessenberg form ( $i = m$ ) or upper triangular form ( $i = 1, 2, \dots, m - 1$ ) by a series of Householder reflections. The second stage is iteration of periodic QR algorithm extended from Francis's algorithm [29] for the standard case ( $m = 1$ ). The convergence of this stage is guaranteed by the "Implicit Q Theorem" [29, 12]. Once the second stage is accomplished, the process of computing eigenvalues is quite simple. If the  $i_{th}$  eigenvalue is real, it is given by the product of all the  $i_{th}$  diagonal elements of matrices  $R_1, R_2, \dots, R_m$ . In practice, the logarithms of magnitudes of these numbers are added, in order to overcome numerical overflows. If the  $i_{th}$  and  $(i + 1)_{th}$  eigenvalues form a complex conjugate pair, all  $[2 \times 2]$  matrices at position  $(i, i + 1)$  on the diagonal of  $R_1, R_2, \dots, R_m$  are multiplied with normalization at each step, and the two complex eigenvalues of the product are obtained. There is no danger of numerical overflow because all these  $[2 \times 2]$  matrices are in the same position and in our applications their elements are of similar order of magnitude.

**4.2. Simultaneous iteration.** The basic idea of simultaneous iteration is implementing QR decomposition in the process of power iteration. Assume all the eigenvalues of  $\mathbf{J}^{(0)}$  are real, without degeneracy, and order them by their magnitude:  $|\Lambda_1| > |\Lambda_2| > \dots > |\Lambda_n|$ , with corresponding normalized unit eigenvectors  $v_1, v_2, \dots, v_n$ . For simplicity, here we have dropped the upper indices of these vectors. An arbitrary initial vector  $\tilde{q}_1 = \sum_{i=1}^n \alpha_i^{(1)} v_i$  will converge to the first eigenvector

$v_1$  after normalization under power iteration of  $\mathbf{J}^{(0)}$ ,

$$\lim_{\ell \rightarrow \infty} \frac{(\mathbf{J}^{(0)})^\ell \tilde{q}_1}{\|\cdot\|} \rightarrow q_1 = v_1.$$

Here  $\|\cdot\|$  denotes the Euclidean norm of the numerator ( $\|x\| = \sqrt{x^\top x}$ ). Let  $\langle a, b, \dots, c \rangle$  represent the space spanned by vector  $a, b, \dots, c$  in  $\mathbb{R}^n$ . Another arbitrary vector  $\tilde{q}_2$  is then chosen orthogonal to subspace  $\langle q_1 \rangle$  by Gram-Schmidt orthonormalization,  $\tilde{q}_2 = \sum_{i=2}^n \alpha_i^{(2)} [v_i - (q_1^\top v_i) q_1]$ . Note that the index starts from  $i = 2$  because  $\langle q_1 \rangle = \langle v_1 \rangle$ . The strategy now is to apply power iteration of  $\mathbf{J}^{(0)}$  followed by orthonormalization in each iteration.

$$\begin{aligned} \mathbf{J}^{(0)} \tilde{q}_2 &= \sum_{i=2}^n \alpha_i^{(2)} [\Lambda_i v_i - \Lambda_1 (q_1^\top v_i) v_1] \\ &= \sum_{i=2}^n \alpha_i^{(2)} \Lambda_i [v_i - (q_1^\top v_i) v_1] + \sum_{i=2}^n \alpha_i^{(2)} (\Lambda_i - \Lambda_1) (q_1^\top v_i) v_1. \end{aligned}$$

The second term in the above expression will disappear after performing Gram-Schmidt orthonormalization to  $\langle q_1 \rangle$ , and the first term will converge to  $q_2 = v_2 - (q_1^\top v_2) v_1$  (not normalized) after a sufficient number of iterations because of the decreasing magnitudes of  $\Lambda_i$ , and we also note that  $\langle v_1, v_2 \rangle = \langle q_1, q_2 \rangle$ . The same argument can be applied to  $\tilde{q}_i$ ,  $i = 3, 4, \dots, n$  as well. When  $\tilde{q}_{j-1}$  converges to  $q_{j-1} = v_{j-1} - \sum_{s=1}^{j-2} (q_s^\top v_{j-1}) q_s$ , we have  $\langle v_1, v_2, \dots, v_{j-1} \rangle = \langle q_1, q_2, \dots, q_{j-1} \rangle$ . Choose an arbitrary vector  $\tilde{q}_j$  perpendicular to subspace  $\langle v_1, v_2, \dots, v_{j-1} \rangle$ :  $\tilde{q}_j = \sum_{i=j}^n \alpha_i^{(j)} [v_i - \sum_{s=1}^{j-1} (q_s^\top v_i) q_s]$ . After one iteration,

$$\mathbf{J}^{(0)} \tilde{q}_j = \sum_{i=j}^n \alpha_i^{(j)} \Lambda_i \left[ v_i - \sum_{s=1}^{j-1} (q_s^\top v_i) q_s \right] + \sum_{i=j}^n \alpha_i^{(j)} \sum_{s=1}^{j-1} (\Lambda_i - \Lambda_s) (q_s^\top v_i) q_s.$$

The second term is a polynomial of  $q_1, q_2, \dots, q_{j-1}$  which is also a polynomial of  $v_1, v_2, \dots, v_{j-1}$ , so it disappears after orthonormalization and the first term will converge to  $q_j = v_j - \sum_{s=1}^{j-1} (q_s^\top v_j) q_s$  (not normalized). In this way, after a sufficient number of iterations,

$$\lim_{\ell \rightarrow \infty} (\mathbf{J}^{(0)})^\ell [\tilde{q}_1, \tilde{q}_2, \dots, \tilde{q}_n] \rightarrow [q_1, q_2, \dots, q_n],$$

where

$$\begin{aligned} q_1 &= v_1, & q_2 &= \frac{v_2 - (v_2^\top q_1) q_1}{\|\cdot\|}, \\ q_3 &= \frac{v_3 - (v_3^\top q_1) q_1 - (v_3^\top q_2) q_2}{\|\cdot\|}, & \dots, & & q_n &= \frac{v_n - \sum_{i=1}^{n-1} (v_n^\top q_i) q_i}{\|\cdot\|}. \end{aligned}$$

Let matrix  $Q_0 = [q_1, q_2, \dots, q_n]$ ; then we have  $\mathbf{J}^{(0)} Q_0 = Q_0 \mathbf{R}^{(0)}$  with  $\mathbf{R}^{(0)}$  an upper triangular matrix because of  $\langle q_1, q_2, \dots, q_i \rangle = \langle v_1, v_2, \dots, v_i \rangle$ , which is just  $\mathbf{J}^{(0)} = Q_0 \mathbf{R}^{(0)} Q_0^\top$  (the Schur decomposition of  $\mathbf{J}^{(0)}$ ). The diagonal elements of  $\mathbf{R}^{(0)}$  are the eigenvalues of  $\mathbf{J}^{(0)}$  in decreasing order.

Numerically, the process described above can be implemented on an arbitrary full rank matrix  $\tilde{Q}_0$  followed by QR decomposition at each step  $J_s \tilde{Q}_{s-1} = \tilde{Q}_s \tilde{R}_s$

with  $s = 1, 2, 3, \dots$  and  $J_{s+m} = J_s$ . For sufficient number of iterations,  $\tilde{Q}_s$  and  $\tilde{R}_s$  converge to  $Q_s$  and  $R_s$  (3.3) for  $s = 1, 2, \dots, n$ , so we achieve (3.4) the periodic Schur decomposition of  $\mathbf{J}^{(k)}$ .

We have thus demonstrated that the simultaneous iteration converges to the periodic Schur decomposition of  $\mathbf{J}_k$  for real non-degenerate eigenvalues. For complex eigenvalue pairs, the algorithm converges in the sense that the subspace spanned by the complex conjugate vector pair converges. So,

$$\mathbf{J}^{(0)}Q_0 = Q_0' \mathbf{R}^{(0)} = Q_0 B \mathbf{R}^{(0)},$$

where  $B$  is a block-diagonal matrix with diagonal elements  $\pm 1$  (corresponding to real eigenvalues) or  $[2 \times 2]$  blocks (corresponding to complex eigenvalue pairs). Absorb  $B$  into  $R_m$ , then  $R_m$  becomes a quasi-upper triangular matrix, and (3.3) still holds.

**5. Eigenvector algorithms.** The eigenvectors of  $\mathbf{J}^{(k)}$  are related to eigenvectors of  $\mathbf{R}^{(k)}$  by orthogonal matrix  $Q_k$  from (3.3). The eigenvector matrix of  $\mathbf{R}^{(k)}$  has the same quasi-upper triangular structure as  $\mathbf{R}^{(k)}$ , so power iteration on an initial arbitrary quasi-upper triangular matrix will generate the eigenvector matrix. This is the basic idea of the first algorithm for generating eigenvectors of  $\mathbf{R}^{(k)}$ , inspired by the algorithm for calculating covariant vectors for an ergodic system [14]. Observation that the first eigenvector of  $\mathbf{R}^{(k)}$  is trivial if it is real,  $\mathbf{e}_1 = (1, 0, \dots, 0)^\top$ , now inspires us to reorder the eigenvalues so that the  $j_{th}$  eigenvalue is in the first diagonal place of  $\mathbf{R}^{(k)}$ ; in this way, the  $j_{th}$  eigenvector is obtained. For both methods, attention should be paid to the complex conjugate eigenvector pairs.

**5.1. Iteration algorithm.** The prerequisite for iteration algorithm is that all the eigenvalues are ordered in an ascending or descending way by their magnitude on the diagonal of  $\mathbf{R}^{(k)}$ . Assume that they are in descending order, which is the outcome of simultaneous iteration; therefore the diagonal elements of  $\mathbf{R}^{(k)}$  are  $\Lambda_1, \Lambda_2, \dots, \Lambda_n$ , with magnitudes from large to small.

**5.1.1. Real eigenvectors.** If the  $i_{th}$  eigenvector of  $\mathbf{R}^{(k)}$  is real, then it has the form  $\mathbf{e}_i = (a_1, a_2, \dots, a_i, 0, \dots, 0)^\top$ . An arbitrary vector whose first  $i$  elements are nonzero  $x = (b_1, b_2, \dots, b_i, 0, \dots, 0)^\top$  is a linear combination of the first  $i$  eigenvectors:  $x = \sum_{j=1}^i \alpha_j \mathbf{e}_j$ . Use it as the initial condition for the power iteration by  $(\mathbf{R}^{(k)})^{-1} = R_{k+1}^{-1} \dots R_m^{-1} R_1^{-1} R_2^{-1} \dots R_k^{-1}$  and after  $\ell$  iterations,

$$(\mathbf{R}^{(k)})^{-\ell} x = \frac{1}{\Lambda_i^\ell} \left( \sum_{j=1}^{i-1} \alpha_j \frac{\Lambda_i^\ell}{\Lambda_j^\ell} v_j + \alpha_i v_i \right).$$

The property we used here is that  $(\mathbf{R}^{(k)})^{-1}$  and  $\mathbf{R}^{(k)}$  have the same eigenvectors but inverse eigenvalues. Power iteration will converge to the  $i_{th}$  eigenvector of  $\mathbf{R}^{(k)}$  if this vector is normalized after each iteration,

$$\lim_{\ell \rightarrow \infty} \frac{(\mathbf{R}^{(k)})^{-\ell} x}{\|\cdot\|} = \mathbf{e}_i.$$

**5.1.2. Complex eigenvector pairs.** For a  $[2 \times 2]$  block on the diagonal of  $\mathbf{R}^{(k)}$ , the corresponding eigenvectors are a complex conjugate pair. Since the two eigenvalues  $\Lambda_i$  and  $\Lambda_{i+1} = \Lambda_i^*$  have the same magnitude, the method needs to be modified in order to tell them apart. If one starts the power iteration with a real



vector, then this vector will rotate. The power iteration still works, in the sense that the subspace spanned by the complex conjugate eigenvector pair converges.

Suppose the  $i_{th}$  and  $(i+1)_{th}$  eigenvectors of  $\mathbf{R}^{(k)}$  form a complex pair. Two arbitrary vectors  $x_1$  and  $x_2$  whose first  $i+1$  elements are non zero can be written as the linear superposition of the first  $i+1$  eigenvectors,

$$x_{1,2} = \left( \sum_{j=1}^{i-1} \alpha_j^{(1,2)} \mathbf{e}_j \right) + \alpha_i^{(1,2)} \mathbf{e}_i + (\alpha_i^{(1,2)} \mathbf{e}_i)^*,$$

where  $(*)$  denotes the complex conjugate. As for the real case, the first  $i-1$  components above will vanish after a sufficient number of iterations. Denote the two vectors at this instant (corresponding to  $x_{1,2}$ ) to be  $X_1$  and  $X_2$  and form matrix  $X = [X_1, X_2]$ . The subspace spanned by  $X_{1,2}$  does not change and  $X$  will be rotated after another iteration,

$$(5.1) \quad (\mathbf{R}^{(k)})^{-1} X = X' = X C,$$

where  $C$  is a  $[2 \times 2]$  matrix which has two complex conjugate eigenvectors  $\mathbf{e}_C$  and  $(\mathbf{e}_C)^*$ . Transformation (5.1) relates the eigenvectors of  $\mathbf{R}^{(k)}$  with those of  $C$ :  $[\mathbf{e}_i, (\mathbf{e}_i)^*] = X[\mathbf{e}_C, (\mathbf{e}_C)^*]$ . In practice, matrix  $C$  can be computed by QR decomposition; let  $X = Q_X R_X$  be the QR decomposition of  $X$ , then  $C = R_X^{-1} Q_X^\top X'$ .

Complex eigenvectors are not uniquely determined in the sense that  $e^{i\theta} \mathbf{e}_i$  is also a eigenvector with the same eigenvalue as  $\mathbf{e}_i$  for an arbitrary angle  $\theta$ , so when comparing results from different eigenvector algorithms, we need a constraint to fix the phase of a complex eigenvector, such as letting the first element be real.

**5.1.3. Shifted power iteration.** In the above, we have implemented power iteration to obtain all the real and complex eigenvectors of matrix  $\mathbf{R}^{(k)}$ . The convergence rate of this pure power iteration algorithm depends on the gap of magnitude among the eigenvalues of  $\mathbf{R}^{(k)}$ , so the performance is relatively poor for systems like Kuramoto-Sivashinsky equation, for which the strongly contracting multipliers (eigenvalues of Jacobian matrix) appear in closely spaced pairs. For a single matrix, inverse iteration [27] is effective to isolate one eigenvalue from the others and thus accelerate the converging process; however, we do not implement it here because of the heavy cost associated with solving linear equation  $((\mathbf{R}^{(k)})^{-1} - sI)y = x$  at one intermediate step of this method, where  $s$  is the shift. Instead, we obtain a better convergence rate by combining the pure power iteration with the ‘shifted power iteration’. The shifted power iteration, is based on the observation that matrix  $(\mathbf{R}^{(k)})^{-1} - sI$  has the same eigenvectors as  $(\mathbf{R}^{(k)})^{-1}$ , but with eigenvalues shifted by an arbitrary number  $s$ , which can be tune to optimize the convergence.

We assume the eigenvalues of  $\mathbf{R}^{(k)}$  are arranged in descending order by magnitude:  $|\Lambda_1| \geq |\Lambda_2| \geq \dots \geq |\Lambda_n|$ . Define  $\Lambda_i = e^{\lambda_i}$  with  $\lambda_i = \mu_i + i\omega_i, \omega \in [0, 2\pi)$ , so  $\mu_i$  represents the magnitude  $|\Lambda_i| = e^{\mu_i}$  of the  $i_{th}$  eigenvalue and  $\omega_i$  distinguishes among positive real ( $\omega_i = 0$ ), negative real ( $\omega_i = \pi$ ) and complex ( $\omega_i \neq 0, \pi$ ). We also assume that  $\mu_i \approx \mu_{i-1}$ , so pure power iteration converges slowly for the  $i_{th}$  eigenvector  $\mathbf{e}_i$ .

The shift power iteration takes different forms for the case  $\Lambda_{i-1}$  is real and  $\Lambda_{i-1}$  is complex. Consider the former case first. In this case,  $e^{\lambda_{i-1}} (\mathbf{R}^{(k)})^{-1} - I$  could be used instead of  $(\mathbf{R}^{(k)})^{-1}$  for power iteration. As the same with the pure power iteration, we start with an arbitrary real vector whose first  $i$  elements are nonzero,

$$(5.2) \quad x = \left( \sum_{j=1}^{i-2} \alpha_j \mathbf{e}_j \right) + \alpha_{i-1} \mathbf{e}_{i-1} + \alpha_i \mathbf{e}_i + (\alpha_i \mathbf{e}_i)^*.$$



This expression is general and the only requirement is that  $a_{i-1}$  and  $\mathbf{e}_{i-1}$  are real. Complex conjugate pair  $\alpha_j \mathbf{e}_j$ ,  $\alpha_{j+1} \mathbf{e}_{j+1} = (\alpha_j \mathbf{e}_j)^*$  may show up in the first term of (5.2), and also  $\alpha_i \mathbf{e}_i + (\alpha_i \mathbf{e}_i)^*$  generates a real vector no matter whether  $\mathbf{e}_i$  is real or complex. After one iteration,

$$\begin{aligned} \left( e^{\lambda_{i-1}} (\mathbf{R}^{(k)})^{-1} - I \right) x &= \sum_{j=1}^{i-2} \alpha_j (e^{\lambda_{i-1}-\lambda_j} - 1) \mathbf{e}_j \\ &\quad + \alpha_i (e^{\lambda_{i-1}-\lambda_i} - 1) \mathbf{e}_i + \alpha_i^* (e^{\lambda_{i-1}-\lambda_i^*} - 1) \mathbf{e}_i^*. \end{aligned}$$

The shift chosen here annihilates term  $a_{i-1} \mathbf{e}_{i-1}$ , but problem arises for the first  $i-2$  terms. They may expand during this shift power iteration, so the combination of pure power iteration and shifted power iteration is required. Since  $e^{\lambda_{i-1}-\lambda_i^*} - 1$  is conjugate to  $e^{\lambda_{i-1}-\lambda_i} - 1$ , we only need to solve

$$\left| e^{(\lambda_i - \lambda_j)N} \cdot \frac{e^{\lambda_{i-1}-\lambda_j} - 1}{e^{\lambda_{i-1}-\lambda_i} - 1} \right| = r_0,$$

where  $r_0$  is the desired convergence rate, we get the number of pure power iterations prior one shifted power iteration,

$$(5.3) \quad N = \max_{j=1}^{i-2} \frac{\ln \left( r_0 \left| \frac{e^{\lambda_{i-1}-\lambda_i} - 1}{e^{\lambda_{i-1}-\lambda_j} - 1} \right| \right)}{\mu_i - \mu_j}.$$

Numerically,  $a_{i-1} \mathbf{e}_{i-1}$  cannot be annihilated completely, but this method still works if the shift chosen here is “closer” to  $e^{\lambda_{i-1}}$  than  $e^{\lambda_i}$ .

When  $\Lambda_{i-1}$  is complex, the situation is a bit more complicated. Now the shifted power iteration takes the form

$$(e^{\lambda_{i-1}} (\mathbf{R}^{(k)})^{-1} - I)(e^{\lambda_{i-1}^*} (\mathbf{R}^{(k)})^{-1} - I) = e^{2\mu_{i-1}} (\mathbf{R}^{(k)})^{-2} - 2e^{\mu_{i-1}} \cos \omega_{i-1} (\mathbf{R}^{(k)})^{-1} + I$$

in order to annihilate both  $a_{i-1} \mathbf{e}_{i-1}$  and  $(a_{i-1} \mathbf{e}_{i-1})^*$ . The number of pure power iteration prior one shifted power iteration can be determined in a similar way,

$$(5.4) \quad N = \max_{j=1}^{i-3} \frac{\ln \left( r_0 \left| \frac{\exp(2\mu_{i-1} - 2\lambda_i) - 2 \cos \omega_{i-1} \exp(\mu_{i-1} - \lambda_i) + 1}{\exp(2\mu_{i-1} - 2\lambda_j) - 2 \cos \omega_{i-1} \exp(\mu_{i-1} - \lambda_j) + 1} \right| \right)}{\mu_i - \mu_j}.$$

**5.2. Reordering algorithm.** The iteration algorithm described above has two drawbacks. First, the rate of convergence depends on the sizes of gaps between the magnitudes of eigenvalues. If a gap is small, the convergence is slow. Even though shifted power iteration can be introduced, the number of prior pure power iterations (5.3), (5.4) may be large for some problems, such as determining the stability of periodic orbits in Kuramoto-Sivashinsky system (discussed in sect. 7 below). Second, we cannot get the eigenvectors of  $\mathbf{R}^{(k)}$  for all  $k \in 0, 1, 2, \dots, m$  at the same time. Although eigenvectors of  $\mathbf{R}^{(k)}$  and  $\mathbf{R}^{(0)}$  are related by (3.5), it is not advisable, as pointed out above, to evolve the eigenvectors of  $\mathbf{R}^{(0)}$  so as to get eigenvectors of  $\mathbf{R}^{(k)}$  because of the noise introduced during this process. Therefore, iteration is needed for each  $k \in 0, 1, 2, \dots, m$ .

There exists a direct algorithm to obtain the eigenvectors of every  $\mathbf{R}^{(k)}$  at once without iteration. The idea is very simple: the eigenvector corresponding to the first

diagonal element of an upper-triangular matrix is  $\mathbf{e}_1 = (1, 0, \dots, 0)^\top$ . By reordering the diagonal elements (or  $[2 \times 2]$  blocks) of  $\mathbf{R}^{(0)}$ , we can find any eigenvector by positioning the corresponding eigenvalue in the first diagonal position. Although in our application only reordering of  $[1 \times 1]$  and  $[2 \times 2]$  blocks is needed, we recapitulate here the general case of reordering two adjacent blocks of a quasi-upper triangular matrix following Granat [17]. Partition  $R_i$  as

$$R_i = \left[ \begin{array}{c|cc|c} R_i^{00} & * & * & * \\ \hline 0 & R_i^{11} & R_i^{12} & * \\ 0 & 0 & R_i^{22} & * \\ \hline 0 & 0 & 0 & R_i^{33} \end{array} \right],$$

where  $R_i^{00}, R_i^{11}, R_i^{22}, R_i^{33}$  have size  $[p_0 \times p_0], [p_1 \times p_1], [p_2 \times p_2]$  and  $[p_3 \times p_3]$  respectively, and  $p_0 + p_1 + p_2 + p_3 = n$ . In order to exchange the middle two blocks ( $R_i^{11}$  and  $R_i^{22}$ ), we construct a non-singular periodic matrix sequence:  $\hat{S}_i$ ,  $i = 0, 1, 2, \dots, m$  with  $\hat{S}_0 = \hat{S}_m$ ,

$$\hat{S}_i = \left[ \begin{array}{c|c|c} I_{p_0} & 0 & 0 \\ \hline 0 & S_i & 0 \\ \hline 0 & 0 & I_{p_3} \end{array} \right],$$

where  $S_i$  is a  $[(p_1 + p_2) \times (p_1 + p_2)]$  matrix, such that  $\hat{S}_i$  transforms  $R_i$  as follows:

$$(5.5) \quad \hat{S}_i^{-1} R_i \hat{S}_{i-1} = \tilde{R}_i = \left[ \begin{array}{c|cc|c} R_i^{00} & * & * & * \\ \hline 0 & R_i^{22} & 0 & * \\ 0 & 0 & R_i^{11} & * \\ \hline 0 & 0 & 0 & R_i^{33} \end{array} \right],$$

which is

$$S_i^{-1} \begin{bmatrix} R_i^{11} & R_i^{12} \\ 0 & R_i^{22} \end{bmatrix} S_{i-1} = \begin{bmatrix} R_i^{22} & 0 \\ 0 & R_i^{11} \end{bmatrix}.$$

The problem is to find the appropriate matrix  $S_i$  which satisfies the above condition. Assume  $S_i$  has form

$$S_i = \begin{bmatrix} X_i & I_{p_1} \\ I_{p_2} & 0 \end{bmatrix},$$

where matrix  $X_i$  has dimension  $[p_1 \times p_2]$ . We obtain periodic Sylvester equation [17]

$$(5.6) \quad R_i^{11} X_{i-1} - X_i R_i^{22} = -R_i^{12}, \quad i = 0, 1, 2, \dots, m.$$

The algorithm to find eigenvectors is based on (5.6). If the  $i_{th}$  eigenvalue of  $\mathbf{R}^{(k)}$  is real, we only need to exchange the first  $[(i-1) \times (i-1)]$  block of  $R_k$ ,  $k = 1, 2, \dots, m$  with its  $i_{th}$  diagonal element. If the  $i_{th}$  and  $(i+1)_{th}$  eigenvalues form a complex pair, then the first  $[(i-1) \times (i-1)]$  block and the following  $[2 \times 2]$  block should be exchanged. Therefore  $X_i$  in (5.6) has dimension  $[p_1 \times 1]$  or  $[p_1 \times 2]$ . In both cases,  $p_0 = 0$ .

**5.2.1. Real eigenvectors.** In this case, matrix  $X_i$  is just a column vector, so (5.6) is equivalent to

$$(5.7) \quad \begin{bmatrix} R_1^{11} & -R_1^{22}I_{p_1} & & & \\ & R_2^{11} & -R_2^{22}I_{p_1} & & \\ & & R_3^{11} & -R_3^{22}I_{p_1} & \\ & & & \ddots & \dots \\ -R_m^{22}I_{p_1} & & & & R_m^{11} \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ \dots \\ X_{m-1} \end{bmatrix} = \begin{bmatrix} -R_1^{12} \\ -R_2^{12} \\ -R_3^{12} \\ \dots \\ -R_m^{12} \end{bmatrix},$$

where  $R_i^{22}$  is the  $(p_1 + 1)_{th}$  diagonal element of  $R_i$ . This is a bordered almost block diagonal matrix, which has dimension  $[(p_1 m) \times (p_1 m)]$ . Gaussian elimination with partial pivoting (GEPP) is used to solve (5.7). Ref. [11] argues that GEPP may fail for some matrices, and other methods such as cyclic reduction or preconditioned conjugate gradients, to name a few, have been proposed [1, 10, 16].

Now we get all vectors  $X_i$  by solving periodic Sylvester equation, but how are they related to the eigenvectors? Defining  $\tilde{\mathbf{R}}_0 = \tilde{R}_m \tilde{R}_{m-1} \cdots \tilde{R}_1$ , we get  $\hat{S}_m^{-1} \mathbf{R}^{(0)} \hat{S}_m = \tilde{\mathbf{R}}_0$  by (5.5). Since  $p_0 = 0$  and  $p_2 = 1$  in (5.5), the first eigenvector of  $\tilde{\mathbf{R}}_0$ , the one corresponding to eigenvalue  $\Pi_{i=1}^m R_i^{22}$  is  $\tilde{e} = (1, 0, \dots, 0)^\top$ . Before normalization, the corresponding eigenvector of  $\mathbf{R}^{(0)}$  is

$$\mathbf{e}_{p_1+1} = \hat{S}_m \tilde{e} = [X_0^\top, 1, 0, 0, \dots, 0]^\top.$$

This is the eigenvector of matrix  $\mathbf{R}^{(0)} = R_m R_{m-1} \cdots R_1$  in (3.4) for  $k = 0$ . For  $\mathbf{R}^{(1)} = R_1 R_m \cdots R_2$ , the corresponding periodic Sylvester equation will be cyclically rotated one row up, which means  $X_1$  will be shifted to the first place in the column vector in (5.7), and thus the corresponding eigenvector of  $\mathbf{R}^{(1)}$  is  $\mathbf{e}_{p_1+1} = [X_1^\top, 1, 0, \dots, 0]^\top$ . The same argument goes for all the following  $\mathbf{R}^{(k)}$ ,  $k = 2, 3, \dots, m-1$ . In conclusion, solution of (5.7) contains the eigenvectors for all  $\mathbf{R}^{(k)}$ ,  $k = 0, 1, \dots, m-1$ .

**5.2.2. Complex eigenvector pairs.** As in the real eigenvalue case, we have  $p_0 = 0$ , but now  $p_2 = 2$ , so matrix  $X_i$  has dimension  $[p_1 \times 2]$ . Using the same notation as ref. [17], let  $v(X_i)$  denote the vector representation of  $X_i$  with the columns of  $X_i$  stacked on top of each other, and let  $A \otimes B$  denote the Kronecker product of two matrices, with the  $(i, j)$ -block element be  $a_{ij}B$ .

Now, the periodic Sylvester equation (5.6) is equivalent to

$$(5.8) \quad \begin{bmatrix} I_2 \otimes R_1^{11} & -(R_1^{22})^\top \otimes I_{p_1} & & & \\ & I_2 \otimes R_2^{11} & -(R_2^{22})^\top \otimes I_{p_1} & & \\ & & I_2 \otimes R_3^{11} & -(R_3^{22})^\top \otimes I_{p_1} & \\ & & & \ddots & \dots \\ -(R_m^{22})^\top \otimes I_{p_1} & & & & I_2 \otimes R_m^{11} \end{bmatrix} \begin{bmatrix} v(X_0) \\ v(X_1) \\ v(X_2) \\ \dots \\ v(X_{m-1}) \end{bmatrix} = \begin{bmatrix} -v(R_1^{12}) \\ -v(R_2^{12}) \\ -v(R_3^{12}) \\ \dots \\ -v(R_m^{12}) \end{bmatrix}.$$

After switching  $R_i^{11}$  and  $R_i^{22}$ , we can get the first two eigenvectors of  $\tilde{\mathbf{R}}_0$  by multiplying the first  $[2 \times 2]$  diagonal blocks of  $\tilde{R}_i$ :  $R^{22} = R_m^{22} R_{m-1}^{22} \cdots R_1^{22}$ . Let the eigenvectors of  $R^{22}$  be  $v$  and  $v^*$  of size  $[2 \times 1]$ , then the corresponding eigenvectors of  $\tilde{\mathbf{R}}_0$  are

$\tilde{e}_1 = (v^\top, 0, 0, \dots, 0)^\top$  and  $\tilde{e}_2 = (\tilde{e}_1)^*$  (the additional zeros make the length of the eigenvectors to be  $n$ ). Therefore, the corresponding eigenvector of  $\mathbf{R}^{(0)}$  is

$$[\mathbf{e}_{p_1+1}, \mathbf{e}_{p_1+2}] = \hat{S}_m[\tilde{e}_1, \tilde{e}_2] = \begin{bmatrix} X_0 \\ I_2 \\ 0 & 0 \\ 0 & 0 \\ \vdots \\ 0 & 0 \end{bmatrix} [v, v^*].$$

For other  $\mathbf{R}^{(k)}$ , the same argument in the real case applies here too, so we obtain all the complex eigenvector pairs for  $\mathbf{R}^{(k)}$ ,  $k = 1, 2, \dots, m$ .

**6. Computational effort and convergence analysis.** In this paper we make no attempt at conducting a strict error analysis of the alternative algorithms presented. However, for practical applications it is important to understand their computational costs.

Periodic eigendecomposition is conducted in two stages: (1) periodic Schur decomposition, and (2) determination of all eigenvectors. In each stage, there are two candidate algorithms, so the efficiency of periodic eigendecomposition depends on the choice of the specific algorithm in each stage.

Periodic QR algorithm and simultaneous iteration are both effective to achieve PRSF for the real eigenvalues, and for complex pairs of eigenvalues. Periodic QR algorithm consists of two stages. First, matrix sequence  $J_m, J_{m-1}, \dots, J_1$  is reduced to Hessenberg-triangular form, with  $J_{m-1}, \dots, J_1$  upper triangular and  $J_m$  upper Hessenberg. It requires  $O(mn)$  Householder reflections in this stage and computational cost associated with each reflection is  $O(n^2)$ , if the transformed matrix is calculated implicitly without forming the Householder matrix [27]. So the overall computational cost of this stage is  $O(mn^3)$ . The second stage is the periodic QR iteration which is a generalization of the standard,  $m = 1$ , case [27].  $O(mn)$  Givens rotations are performed in each iteration with overall computational cost of  $O(mn^2)$ . Though the computational effort in each iteration in the second stage is less than that in the first stage, the number of iterations in the second stage is usually far more than the dimension of matrices involved. In this sense, the second stage is the heavy part of periodic QR algorithm. On the other hand, simultaneous iteration conducts one QR decomposition  $O(n^3)$  and  $m$  matrix-matrix multiplication  $O(n^3)$  in each iteration, giving a total computational cost of  $O(mn^3)$ . The convergence of either algorithm depends linearly on the ratio of adjacent eigenvalues of  $\mathbf{R}^{(0)}$ :  $|\Lambda_i|/|\Lambda_{i+1}|$  without shift [12]. Therefore the ratio of costs is of the order  $O(mn^3)/O(mn^2) \approx O(n)$ , implying that the periodic QR algorithm is much cheaper than the simultaneous iteration if the dimension of matrices involved is large enough.

The second stage of periodic eigendecomposition is to find all the eigenvectors of  $\mathbf{J}^{(k)}$  via quasi-upper triangular matrices  $\mathbf{R}^{(k)}$ . The first candidate is the combination of power iteration and shifted power iteration. The computational cost of one iteration for the  $i_{th}$  eigenvector is  $O(mi^2)$ . The second candidate, reordering algorithm, relies on an effective method to solve periodic Sylvester equation (5.6). For example, Gaussian elimination with partial pivoting (GEPP) is suitable for well conditioned matrix (5.7) and (5.8) with computational cost of  $O(mn^2)$ . On the other hand, the iteration algorithm, as pointed out earlier, could not produce the eigenvectors of  $\mathbf{R}^{(k)}$  for all  $k = 1, 2, \dots, m$  accurately in the same time due to the noise introduced during

the transformation process (3.4), especially when the magnitudes of eigenvalues span a large range. In contrast, the reordering algorithm is not iterative and it gives all the eigenvectors simultaneously.

In summary, if we just consider the computational effort, the combination of periodic QR algorithm and reordering algorithm is the best choice for periodic eigen-decomposition.

**7. Application to Kuramoto-Sivashinsky equation.** As an example, we focus on the one-dimensional Kuramoto-Sivashinsky equation

$$(7.1) \quad u_t + \frac{1}{2}(u^2)_x + u_{xx} + u_{xxxx} = 0, \quad x \in [0, L]$$

on a periodic spatial domain of size  $L$ . The preperiodic orbit  $\overline{pp}$  of period  $T_{pp} = 10.25$ , and relative periodic orbit  $\overline{rp}$  of period  $T_{rp} = 16.3$  that we study here are described in ref. [6], where the domain size has been set to  $L = 22$ , large enough to exhibit complex spatiotemporal dynamics. Periodic boundary condition enables us to transform this partial differential equation into a set of ODEs in Fourier space, and in our computations discrete  $N = 64$  Fourier transform is used,

$$a_k(t) = \mathcal{F}[u]_k = \frac{1}{N} \sum_{n=0}^{N-1} u(x_n, t) e^{-iq_k x_n}, \quad u(x_n, t) = \mathcal{F}^{-1}[a]_k = \sum_{k=-N/2+1}^{N/2} a_k(t) e^{iq_k x_n},$$

where  $q_k = 2\pi k/L$ , and the coefficients are complex,  $a_k = b_k + ic_k$ . The transform of differentiation of  $u(x, t)$  is given by

$$\mathcal{F} \left[ \frac{\partial^\nu u}{\partial x^\nu} \right]_k = \begin{cases} (iq_k)^\nu \mathcal{F}[u]_k, & \text{otherwise} \\ 0 & \text{if } \nu \text{ is odd and } k=N/2. \end{cases}$$

Here the odd derivative of the  $N/2$  mode at the grid points is set to zero separately in order to eliminate the asymmetry of the highest wave number in the definition of inverse discrete Fourier transform [26]. Applying discrete Fourier transform to (7.1) we obtain

$$(7.2a) \quad \dot{a}_k = (q_k^2 - q_k^4) a_k - i \frac{q_k}{2} \mathcal{F}[(\mathcal{F}^{-1}[a])^2]_k, \quad k = -N/2 + 1, \dots, N/2 - 1$$

$$(7.2b) \quad \dot{a}_{N/2} = (q_{N/2}^2 - q_{N/2}^4) a_{N/2}.$$

In our implementation of the code we use Fast Fourier Transform packages, such as Matlab `fft()` function (note, however, that Matlab `fft()` orders wave numbers as  $k = 0, 1, 2, \dots, N-1$ , which is mapped to  $k = 0, 1, \dots, N/2-1, N/2, -N/2+1, \dots, -1$  in Kuramoto-Sivashinsky system).

Since  $u(x, t)$  is real,  $a_k(t) = a_{-k}^*(t)$ ; thus only half of the Fourier modes are independent. As  $\dot{a}_0 = 0$  from (7.2a), we can set  $a_0 = 0$ . It follows from (7.2b) that  $a_{N/2}$  is decoupled from other modes and it can be set to zero as well. Thus then the number of independent variables is  $N - 2$ ,

$$(7.3) \quad \hat{u} = (b_1, c_1, b_2, c_2, \dots, b_{N/2-1}, c_{N/2-1})^\top.$$

This is the ‘state space’ in the discussion that follows. ETDRK4 scheme [18, 4] is implemented to integrate (7.2a). The combination of periodic QR algorithm and reordering algorithm is used to obtain all exponents and eigenvectors. In addition,

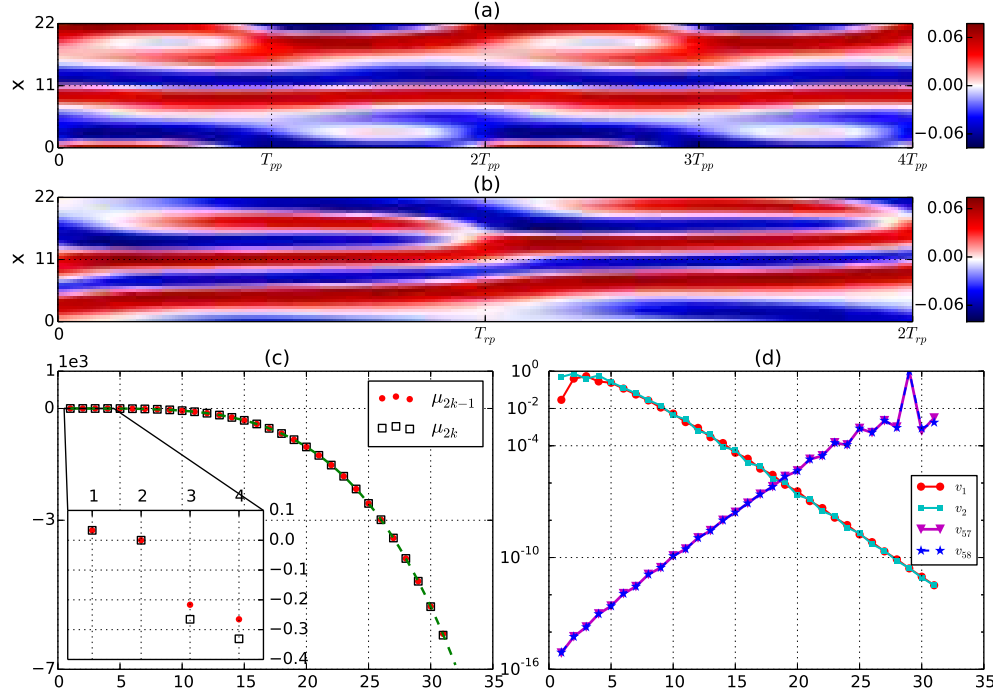


FIG. 1. (Color online) (a) Preperiodic orbit  $\overline{pp}$  and (b) relative periodic orbit  $\overline{rp}$  in the full state space for total time  $4T_{pp}$  and  $2T_{rp}$ , respectively. The phase shift for  $\overline{rp}$  after one prime period  $\simeq -2.863$ . (c) The real parts of Floquet exponents paired for a given  $k$  as  $(k, \mu_{2k-1})$  and  $(k, \mu_{2k})$ , for  $\overline{pp}$  with truncation number  $N = 64$ . The dashed line (green) is  $q_k^2 - q_k^4$ , with  $x$ -axis the indices of Fourier modes  $k = 1, 2, \dots, N/2 - 1$ . The inset is a magnification of the region containing the 8 leading entangled modes. As can be seen in table 1, for modes that follow,  $k \geq 5$ , the exponents are much smaller, in agreement with the expected separation into entangled and isolated modes of ref. [32]. (d) The magnitudes of the Fourier components  $|a_k| = (b_k^2 + c_k^2)^{1/2}$  of the 1st, the 2nd, the 57th and 58th Floquet vectors  $\mathbf{e}_k$  for  $\overline{pp}$  at initial time  $t = 0$ , truncation number  $N = 64$ . For entangled modes the first 4 Fourier are comparable in magnitude. For the  $k_{th}$  isolated modes pair, the amplitude is concentrated on  $k_{th}$  Fourier mode. The  $x$ -axis is labeled by the Fourier mode indices. Only the  $k > 0$  part is shown, the negative  $k$  follow by reflection.

Gaussian elimination with partial pivoting (GEPP) is stable for (5.7) and (5.8) if the time step in Kuramoto-Sivashinsky integrator is not too large, as GEPP only uses addition and subtraction operations.

Kuramoto-Sivashinsky equation is equivariant under reflection and space translation:  $-u(-x, t)$  and  $u(x + l, t)$  are also solutions if  $u(x, t)$  is a solution, which corresponds to equivariance of (7.3) under group operation  $R = \text{diag}(-1, 1, -1, 1, \dots)$  and  $g(l) = \text{diag}(r_1, r_2, \dots, r_{N/2-1})$ , where

$$r_k = \begin{pmatrix} \cos(q_k l) & -\sin(q_k l) \\ \sin(q_k l) & \cos(q_k l) \end{pmatrix}, \quad k = 1, 2, \dots, N/2 - 1.$$

There are three types of recurrent orbits in Kuramoto-Sivashinsky system: periodic orbits in the  $b_k = 0$  invariant antisymmetric subspace, preperiodic orbits which are self-dual under the reflection, and relative periodic orbits with a shift along group orbit after one period. As shown in ref. [6], the first type is absent for domains as small as  $L = 22$ , and thus we focus on the last two types of orbits. For preperiodic

TABLE 1

The first 10 and last four real Floquet exponents and Floquet multiplier phases,  $\Lambda_i = \exp(T\mu_i \pm i\theta_i)$ , for orbits  $\overline{pp}$  and  $\overline{rp}$ , respectively.  $\theta_i$  column lists either the phase, if the Floquet multiplier is complex, or '-1' if the multiplier is real, but inverse hyperbolic. Truncation number  $N = 32$ . The 8 leading exponents correspond to the entangled modes: note the sharp drop in the value of the 9<sub>th</sub> and subsequent exponents, corresponding to the isolated modes.

$\overline{pp}$			$\overline{rp}$		
$i$	$\mu_i$	$\theta_i$	$i$	$\mu_i$	$\theta_i$
1,2	0.033209	$\pm 2.0079$	1	0.32791	
3	-2.0317e-14		2	5.0352e-09	
4	-2.4267e-09	-1	3	-1.2399e-08	
5	-0.21637		4	-0.13214	-1
6,7	-0.26524	$\pm 2.6205$	5,6	-0.28597	$\pm 2.7724$
8	-0.33073	-1	7	-0.36242	
9	-1.9605		8	-0.32821	-1
10	-1.9676	-1	9,10	-1.9617	$\pm 2.2411$
...	...	...	...	...	...
27	-239.52		...	...	...
28	-239.22	-1	27,28	-239.41	$\pm 0.88159$
29	-307.47	-1	29	-313.98	
30	-332.74		30	-323.41	

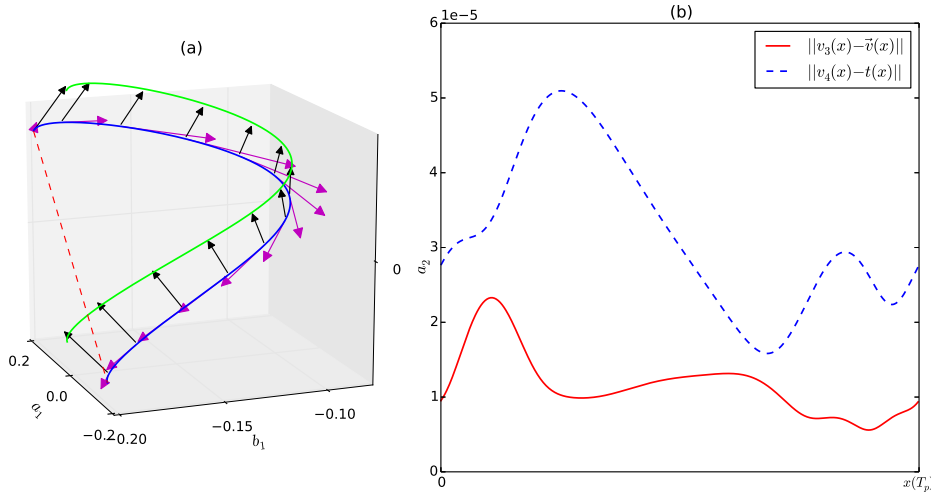


FIG. 2. (Color online) Marginal vectors and the associated errors. (a)  $\overline{pp}$  for  $T_{pp}$  projected onto  $[a_1, b_1, a_2]$  subspace (blue curve), and its counterpart (green line) generated by a small group transformation  $g(\ell)$  by  $\ell$ , here arbitrarily set to  $\ell = L/(20\pi)$ . Magenta and black arrows represent the first and the second marginal Floquet vectors  $\mathbf{e}_3(x)$  and  $\mathbf{e}_4(x)$  along the prime orbit. (b) The solid red curve is the magnitude of the difference between  $\mathbf{e}_3(x)$  and the velocity field  $\vec{v}(x)$  along the orbit, and blue dashed curve is the difference between  $\mathbf{e}_4(x)$  and the group tangent  $t(x) = \mathbf{T}x$ .

orbits  $\hat{u}(0) = R\hat{u}(T_p)$ , we only need to evolve the system for a prime period  $T_p$  which is half of the whole period, with the Floquet matrix given by  $J_p(x) = RJ^{T_p}(x)$ . A relative periodic orbit,  $\hat{u}(0) = g_p\hat{u}(T_p)$ , returns after one period  $\hat{u}(T_p)$  to the initial state upon the group transform  $g_p = g(l_p)$ , so the corresponding Floquet matrix is  $J_p(x) = g_p J^{T_p}(x)$ . Here we show how our periodic eigendecomposition works by applying it to one representative preperiodic orbit  $\overline{pp}$  and one relative periodic orbit  $\overline{rp}$ .

At each repeat of the period  $T_{pp}$ ,  $\overline{pp}$  is invariant under reflection along  $x = L/2$ ,



figure 1 (a), and  $\overline{r\overline{p}}$  has a shift along the  $x$  direction as time goes on, figure 1 (b). Since  $\overline{p\overline{p}}$  and  $\overline{r\overline{p}}$  are both time invariant and equivariant under  $\text{SO}(2)$  group transformation  $g(l)$ , there should be two marginal Floquet exponents, corresponding to the velocity field  $v(x)$  and group tangent  $t(x) = \mathbf{T}x$  respectively, where  $\mathbf{T}$  is the generator of  $\text{SO}(2)$  rotation:

$$\mathbf{T} = \text{diag}(t_1, t_2, \dots, t_{N/2-1}), \quad t_k = \begin{pmatrix} 0 & -q_k \\ q_k & 0 \end{pmatrix}.$$

Table 1 shows that the  $2_{nd}$  and  $3_{rd}$ , respectively  $3_{rd}$  and  $4_{th}$  exponents of  $\overline{r\overline{p}}$ , respectively  $\overline{p\overline{p}}$ , are marginal, with accuracy as low as  $10^{-8}$ , to which the inaccuracy introduced by the error in the closure of the orbit itself also contributes.

We have noted above that the group property of Jacobian matrix multiplication (2.3) enables us to factorize  $\mathbf{J}^{(k)}$  into a product of short-time matrices with matrix elements of comparable magnitudes. In practice, caution should be exercised when trying to determine the optimal number of time increments that the orbit should be divided into. If the number of time increments  $m$  is too large, then, according to the estimates of sect. 6, the computation may be too costly. If  $m$  is too small, then the elements of Jacobian matrix corresponding to the corresponding time increment may range over too many orders of magnitude, causing periodic eigendecomposition to fail to resolve the most contracting Floquet vector along the orbit. One might also vary the time step according to the velocity at a give point on the orbit. Here we have determined satisfactory  $m$ 's by numerical experimentation. We find that it suffices to divide the preperiodic orbit  $\overline{p\overline{p}}$  of period  $T_{pp} \simeq 10.253$  into 500 time increments, and the relative periodic orbit  $\overline{r\overline{p}}$  of period  $T_{rp} \simeq 16.314$  into 820 time increments. Table 1 and figure 1 (c) then show that periodic Schur decomposition is capable of resolving Floquet multipliers differing by thousands of orders to machine accuracy: when  $N = 64$ , the smallest Floquet multiplier for  $\overline{p\overline{p}}$  is  $|\Lambda_{62}| \simeq e^{-6083.72}$ . For sufficiently large index  $k$ , the velocity field of (7.2a) is dominated by the linear term  $(q_k^2 - q_k^4)a_k$ ; therefore these isolated modes are decoupled from the leading entangled modes and are expected to contract with rate  $\approx q_k^2 - q_k^4$ . Figure 1 (c) and (d) illustrate this. For large wave number  $k$ , the real parts of Floquet exponents  $(\mu_{2k-1}, \mu_{2k})$  lie on curve  $q_k^2 - q_k^4$ , and the associated Floquet vectors  $(\mathbf{e}_{2k-1}, \mathbf{e}_{2k})$  peak at  $k$ , indicating their decoupling from other modes. Since these higher modes are isolated and have very large negative Floquet exponents, a perturbation along the corresponding eigen-direction disappears instantaneously without affecting other modes. In this sense, the truncation numbers chosen here are large enough to describe the long-time dynamics in Kuramoto-Sivashinsky system.

The two marginal directions have a simple geometrical interpretation. Figure 2 (a) depicts the two marginal vectors of  $\overline{p\overline{p}}$  projected onto the subspace spanned by  $[a_1, b_1, a_2]$  (the real, imaginary parts of the first mode and the real part of the second Fourier mode). The first marginal eigen-direction (the  $3_{rd}$  Floquet vector in table 1) is aligned with the velocity field along the orbit, and the second marginal direction (the  $4_{th}$  Floquet vector) is aligned with the group tangent. The numerical difference between the unit vectors along these two marginal directions and the corresponding physical directions is shown in figure 2 (b). The difference is larger than the accuracy of the hyperbolic Floquet vectors, of order of  $10^{-5}$ . Both the error of calculating the velocity field, the orbit itself and the error associated with the periodic eigendecomposition contribute to this.

As shown in table 1, for an preperiodic orbit, such as  $\overline{p\overline{p}}$ , the trajectory tangent and the group tangent have eigenvalue  $+1$  and  $-1$  respectively, and are thus distinct.

However, the two marginal directions are degenerate for an relative periodic orbit, such as  $\overline{rp}$ . So these two directions are not fixed, but the plane that they span is uniquely determined. Figure 3 shows the velocity field and group tangent along orbit  $\overline{rp}$  indeed lie in the subspace spanned by these two marginal directions.

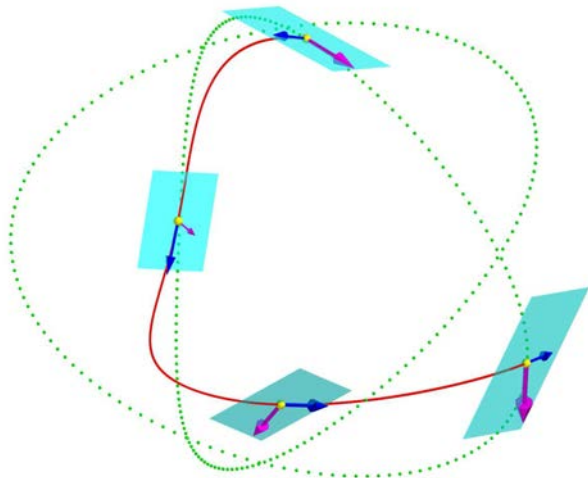


FIG. 3. (Color online) Projection of relative periodic orbit  $\overline{rp}$  onto the Fourier modes subspace  $[b_2, c_2, b_3]$  (red curve). The dotted curve (lime) is the group orbit connecting the initial and final points. Blue and magenta arrows represent the velocity field and group tangent along the orbit, respectively. Two-dimensional planes (cyan) are spanned by the two marginal Floquet vectors at each point (yellow) along the orbit.

**8. Conclusion and future work.** Periodic eigendecomposition, an implementation of the periodic Schur decomposition, is introduced here and its effectiveness demonstrated for linearized dynamics of Kuramoto-Sivashinsky system. As we contemplate applying the method to study of orbits of much longer periods, as well as to the study high-dimensional, numerically exact time-recurrent unstable solutions of the full Navier-Stokes equations [13], we anticipate the need for optimizing and parallelizing such algorithms.

#### REFERENCES

- [1] P. AMODIO, J. R. CASH, G. ROUSSOS, R. W. WRIGHT, G. FAIRWEATHER, I. GLADWELL, G. L. KRAUT, AND M. PAPRZYCKI, *Almost block diagonal linear systems: sequential and parallel solution techniques, and applications*, Numer. Linear Algebra Appl., 7 (2000), pp. 275–317.
- [2] A. BOJANCZYK, G. H. GOLUB, AND P. VAN DOOREN, *The periodic Schur decomposition. Algorithms and applications*, in Proc. SPIE Conference, vol. 1770, 1992, pp. 31–42.
- [3] P. CONSTANTIN, C. FOIAS, B. NICOLAENKO, AND R. TEMAM, *Integral Manifolds and Inertial Manifolds for Dissipative Partial Differential Equations*, Springer, New York, 1989.
- [4] S. M. COX AND P. C. MATTHEWS, *Exponential time differencing for stiff systems*, J. Comput. Phys., 176 (2002), pp. 430–455.
- [5] P. CVITANOVIĆ, R. ARTUSO, R. MAINIERI, G. TANNER, AND G. VATTAY, *Chaos: Classical and Quantum*, Niels Bohr Inst., Copenhagen, 2014. [ChaosBook.org](http://ChaosBook.org).
- [6] P. CVITANOVIĆ, R. L. DAVIDCHACK, AND E. SIMINOS, *On the state space geometry of the Kuramoto-Sivashinsky flow in a periodic domain*, SIAM J. Appl. Dyn. Syst., 9 (2010), pp. 1–33. [arXiv:0709.2944](https://arxiv.org/abs/0709.2944).
- [7] J.-P. ECKMANN, S. O. KAMPHORST, D. RUELLE, AND S. CILIBERTO, *Liapunov exponents from time series*, Phys. Rev. A, 34 (1986), pp. 4971–4979.

- [8] J.-P. ECKMANN AND D. RUELLE, *Ergodic theory of chaos and strange attractors*, Rev. Mod. Phys., 57 (1985), p. 617.
- [9] S. V. ERSHOV AND A. B. POTAPOV, *On the concept of stationary Lyapunov basis*, Physica D, 118 (1998), pp. 167–198.
- [10] G. FAIRWEATHER AND I. GLADWELL, *Algorithms for almost block diagonal linear systems*, SIAM Rev., 46 (2004), pp. 49–58.
- [11] L. FOSTER, *Gaussian elimination with partial pivoting can fail in practice*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 1354–1362.
- [12] J. G. F. FRANCIS, *The QR transformation: A unitary analogue to the LR transformation. I*, Comput. J., 4 (1961), pp. 265–271.
- [13] J. F. GIBSON, J. HALCROW, AND P. CVITANOVIĆ, *Visualizing the geometry of state-space in plane Couette flow*, J. Fluid Mech., 611 (2008), pp. 107–130. [arXiv:0705.3957](#).
- [14] F. GINELLI, H. CHATÉ, R. LIVI, AND A. POLITI, *Covariant Lyapunov vectors*, J. Phys. A, 46 (2013), p. 254005. [arXiv:1212.3961](#).
- [15] F. GINELLI, P. POGGI, A. TURCHI, H. CHATÉ, R. LIVI, AND A. POLITI, *Characterizing dynamics with covariant Lyapunov vectors*, Phys. Rev. Lett., 99 (2007), p. 130601. [arXiv:0706.0510](#).
- [16] R. GRANAT, I. JONSSON, AND B. KÄGSTRÖM, *Recursive blocked algorithms for solving periodic triangular Sylvester-type matrix equations*, in Proc. 8th Intern. Conf. Applied Parallel Computing: State of the Art in Scientific Computing, PARA'06, 2007, pp. 531–539.
- [17] R. GRANAT AND B. KÄGSTRÖM, *Direct eigenvalue reordering in a product of matrices in periodic Schur form*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 285–300.
- [18] A.-K. KASSAM AND L. N. TREFETHEN, *Fourth-order time stepping for stiff PDEs*, SIAM J. Sci. Comput., 26 (2005), pp. 1214–1233.
- [19] K. LUST, *Improved numerical Floquet multipliers*, Internat. J. Bifur. Chaos Appl. Sci. Engrg., 11 (2001), pp. 2389–2410.
- [20] A. LYAPUNOV, *Problème général de la stabilité du mouvement*, Ann. of Math. Studies, 17 (1977). Russian original Kharkow, 1892.
- [21] V. I. OSELEDEC, *A multiplicative ergodic theorem. Liapunov characteristic numbers for dynamical systems*, Trans. Moscow Math. Soc., 19 (1968), pp. 197–221.
- [22] A. POLITI, F. GINELLI, S. YANCHUK, AND Y. MAISTRENKO, *From synchronization to Lyapunov exponents and back*, Physica D, 224 (2006), p. 90. [arXiv:nlin/0605012](#).
- [23] A. POLITI, A. TORCINI, AND S. LEPRI, *Lyapunov exponents from node-counting arguments*, J. Phys. IV, 8 (1998), p. 263.
- [24] K. A. TAKEUCHI, F. GINELLI, AND H. CHATÉ, *Lyapunov analysis captures the collective dynamics of large chaotic systems*, Phys. Rev. Lett., 103 (2009), p. 154103. [arXiv:0907.4298](#).
- [25] J.-L. THIFFEAULT, *Derivatives and constraints in chaotic flows: asymptotic behaviour and a numerical method*, Physica D, 172 (2002), pp. 139–161. [arXiv:nlin/0101012](#).
- [26] L. N. TREFETHEN, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2000.
- [27] L. N. TREFETHEN AND D. BAU, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
- [28] A. TREVISAN AND F. PANCOTTI, *Periodic orbits, Lyapunov vectors, and singular vectors in the Lorenz system*, J. Atmos. Sci., 55 (1998), p. 390.
- [29] D. S. WATKINS, *Francis's algorithm*, Amer. Math. Monthly, 118 (2011), pp. 387–403.
- [30] A. WOLF, J. B. SWIFT, H. L. SWINNEY, AND J. A. VASTANO, *Determining Lyapunov exponents from a time series*, Physica D, 16 (1985), pp. 285–317.
- [31] C. L. WOLFE AND R. M. SAMELSON, *An efficient method for recovering Lyapunov vectors from singular vectors*, Tellus A, 59 (2007), pp. 355–366.
- [32] H.-L. YANG, K. A. TAKEUCHI, F. GINELLI, H. CHATÉ, AND G. RADONS, *Hyperbolicity and the effective dimension of spatially-extended dissipative systems*, Phys. Rev. Lett., 102 (2009), p. 074102. [arXiv:0807.5073](#).