Eur. Phys. J. B (2014) 87: 274
DOI: 10.1140/epjb/e2014-50401-1

THE EUROPEAN PHYSICAL JOURNAL B

Regular Article

Continuous pole placement method for time-delayed feedback controlled systems

Viktoras Pyragas[a] and Kestutis Pyragas

Center for Physical Sciences and Technology, A. Goštauto 11, LT-01108 Vilnius, Lithuania

Received 14 June 2014 / Received in final form 12 October 2014
Published online 19 November 2014 –

Abstract. Continuous pole placement method is adapted to time-periodic states of systems with time delay. The method is applied for finding an optimal control matrix in the problem of stabilization of unstable periodic orbits of dynamical systems via time-delayed feedback control algorithm. The optimal control matrix ensures the fastest approach of a perturbed system to the stabilized orbit. An application of the pole placement method to systems with time delay meets a fundamental problem, since the number of the Floquet exponents is infinity, while the number of control parameters is finite. Nevertheless, we show that several leading Floquet exponents can be efficiently controlled. The method is numerically demonstrated for the Lorenz system, which until recently has been considered as a system inaccessible for the standard time-delayed feedback control due to the odd-number limitation. The proposed optimization method is also adapted for an extended time-delayed feedback control algorithm and numerically demonstrated for the Rössler system.

1 Introduction

The research devoted to chaos control, i.e. the stabilization of unstable periodic orbits (UPOs) embedded in chaotic attractors, is one of the most active fields in applied nonlinear science [1]. Among the chaos control methodologies, time-delayed feedback control (TDFC) method has been receiving considerable attention since it was proposed in reference [2]. The TDFC does not require a reference signal corresponding to the desired orbit; here the control signal is formed from a difference between the current state of the system and a time delayed version of itself. The method is noninvasive in the sense that the control force vanishes when the target orbit is reached. The TDFC algorithm has been successfully implemented in quite diverse experimental systems and the number of modifications of the algorithm have been proposed (see Ref. [3] for the review up to 2006). The most important modification essentially improving the performance of the algorithm is the extended TDFC (ETDFC) [4,5]. Recent modifications include the TDFC with variable and distributed delays [6,7], the TDFC with improved basins of attraction of the stabilized states [8–10] as well as the TDFC with an adaptive tuning of the delay time [11] and control gain [12]. Significant practical applications of the TDFC algorithm have been recently reported for an atomic force microscope [13] and for an experimental analysis of bifurcations [14,15]. In addition, we refer to a few very recent proposals for the application of the TDFC in nonlinear optical systems [16], in electrodynamic tethers [17], in a direct-current bus system [18], in the microelectromechanical systems [19], in traffic flows [20], in neural networks [21], in liquid crystals [22], and in spatially extended nonequilibrium systems [23].

The theory of TDFC is difficult because one has to deal with delay-differential equations, the state space of which is infinite-dimensional. One of the most discussed theoretical problems is so-called odd number limitation of TDFC first pointed out by Nakajima [24]. Nakajima's theorem states that UPOs with an odd number of real Floquet multipliers (FMs) larger than unity cannot be stabilized by time-delayed feedback control. The limitation seemed to be supported by experimental and numerical evidence and a number of various modifications have been proposed in order to bypass the limitation [8,9,25–28]. However, Fiedler et al. [29] have refuted the limitation for autonomous systems by presenting a simple counterexample – the delayed feedback controlled system close to a subcritical Hopf bifurcation. Recently the corrected version of Nakajima's theorem has been presented [30,31] and an algorithm of control design for UPOs far away from the Hopf bifurcation has been proposed [32].

However, no attention was paid to the problem of optimization of the control matrix. The aim of this paper is to fill this gap. Here by optimization we mean the design of the control law that ensures the fastest approach of the controlled system to the stabilized orbit. In real-world application, speed of convergence may be of crucial importance. For example, if a robot is controlled by stabilizing periodic orbits in a chaotic attractor [33], the time it needs to react to a changing environment is bounded by the time the system needs to converge to a periodic orbit of a given

[a] e-mail: viktpy@pfi.lt

period. Hence, in practice, one desires to tune the control parameters such that the convergence time to the desired periodic state is minimized. Note that the optimization problem formulated here differs from that considered in reference [12], where we looked for the optimal feedback gain to minimize the mean square of the control signal in the presence of noise.

Generally, the convergence time in nonlinear systems depends on the initial conditions and thus the problem is highly nontrivial. Hinz et al. [34] demonstrated such a dependence for a simple model of time-delayed feedback control of a fixed point. Here we restrict ourselves by consideration of the convergence time in the framework of linear theory, which is valid for the states in the vicinity of the periodic orbit and is independent of initial conditions. To achieve our goal we exploit the pole-placement method. Initially this algorithm was elaborated for time-invariant systems described by ordinary differential equations [35] and later on extended for steady states of linear time-invariant systems with time-delay [36]. Here we adapt the pole placement method for the periodic states of systems controlled by TDFC and ETDFC algorithms. Note that the proposed algorithm is model-dependent since we need to know the system equations in order to apply the subsequent techniques.

Several optimization algorithms were developed for the stabilization of fixed points in time-delay systems. The method based on the gradient sampling algorithm [37] that minimizes directly the non-smooth spectral abscissa function has been proposed in reference [38]. An eigenvalue optimization approach using continuation tools [39] has been applied to analyze the stabilizability of fixed point via TDFC in two-dimensional systems [40]. Both algorithms use an analytical transcendental characteristic equation for computation of eigenvalues of time-delay system. Unfortunately, such an analytical equation for computation of Floquet exponents of periodic orbits under TDFC is not available and thus these algorithms are not applicable to our problem. Predictive feedback chaos control for optimal convergence speed has been developed in reference [41], however, the consideration was restricted to time-discrete systems.

The rest of the paper is organized as follows. In Section 2 we formulate the problem and describe the pole placement algorithm for the TDFC systems. Then we present a numerical demonstration of our algorithm for the Lorenz system [42]. In Section 3 we extend the algorithm for the ETDFC systems and demonstrate it numerically for the Rössler system [43]. Lastly, we finish the paper with conclusions presented in Section 4.

2 Pole placement for the TDFC systems

2.1 Problem formulation

We start from a general problem of stabilization of unstable periodic orbits in dynamical system

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t)] + \mathbf{b}u_{in}(t), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the state vector of the system, $\mathbf{f}[\mathbf{x}]$ describes the vector field of the free system, $\mathbf{b}$ is the $d \times m$ input matrix and $\mathbf{u}_{in} \in \mathbb{R}^m$ is the input variable. We suppose that the free system $\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t)]$ has an unstable τ-periodic orbit $\mathbf{x}(t) = \boldsymbol{\xi}(t) = \boldsymbol{\xi}(t+\tau)$. To stabilize this orbit we use the time-delayed feedback controller of the form

$$\mathbf{u}_{in}(t) = -\kappa \mathbf{k}^T[\mathbf{x}(t) - \mathbf{x}(t-\tau)], \tag{2}$$

where $\mathbf{k}$ is a constant $d \times m$ control matrix and κ is a scalar feedback gain that defines the strength of the control force. In what follows we use this parameter to scan the dependence of the Floquet exponents (FEs) of the controlled orbit on the feedback strength for fixed entries of the control matrix.

Linearizing the controlled system (1) around the target orbit $\boldsymbol{\xi}(t)$, we obtain

$$\delta\dot{\mathbf{x}}(t) = D\mathbf{f}[\boldsymbol{\xi}(t)]\delta\mathbf{x}(t) - \kappa\mathbf{b}\mathbf{k}^T[\delta\mathbf{x}(t) - \delta\mathbf{x}(t-\tau)], \tag{3}$$

where $\delta\mathbf{x}(t) = \mathbf{x}(t) - \boldsymbol{\xi}(t)$ is a small deviation from the UPO. Following the Floquet theory, we express the vector of deviation as $\delta\mathbf{x}(t) = \mathbf{u}(t)e^{\lambda t}$, where $\mathbf{u}(t) = \mathbf{u}(t+\tau)$ is the Floquet function having the same period as the Jacobi matrix, and λ is the corresponding Floquet exponent (FE). Inserting this ansatz in (3), we get a boundary value problem (BVP)

$$\dot{\mathbf{u}}(t) = [\mathbf{A}(t) - \lambda\mathbf{I}]\mathbf{u}(t) - \kappa\mathbf{K}\mathbf{u}(t)\left[1 - e^{-\lambda\tau}\right] \tag{4}$$

with the boundary condition $\mathbf{u}(0) = \mathbf{u}(\tau)$. Here we have used the notation $\mathbf{A}(t) \equiv D\mathbf{f}[\boldsymbol{\xi}(t)]$ for the Jacobian, $\mathbf{I}$ for the d-dimensional identity matrix and

$$\mathbf{K} = \mathbf{b}\mathbf{k}^T \tag{5}$$

for the product of the input and control matrices. The matrix $\mathbf{K}$ represents a square matrix of $d \times d$ dimensions.

Since we deal with time-delay system, the solution of equation (4) results in infinite number of FEs λ_i, $i = 1, \ldots, \infty$. Among these FEs there always exists the trivial FE $\lambda = 0$ associated with the perturbations along the periodic orbit. The TDFC is successful if the real parts of all nontrivial FEs become negative. Suppose, that for the given input matrix $\mathbf{b}$ we seek to optimize the components of the control matrix $\mathbf{k}$ in such a way as to achieve the fastest approach to the stabilized orbit, i.e., to make the minimum of the leading FE as deep as possible.

The problem is nontrivial, since the number of FEs is infinite while the number of the control parameters in the matrix $\mathbf{k}$ is finite. For this aim we adapt the pole placement method developed for the linear time-invariant systems with time-delay [36] to the case of periodic time-delay systems. For the computation of FEs of the TDFC systems we use a MATLAB package DDE-BIFTOOL [44].

2.2 Description of the algorithm

The idea behind our algorithm is to reduce the real part of the leading FE (the spectral abscissa function) in a quasi-continuous way by applying small changes to the control

matrix, while monitoring the other FEs with the smaller real part. The method is based on the continuity of the FEs with respect to the components of the control matrix. As shown in reference [38], the spectral abscissa function is not everywhere differentiable. Moreover, discontinuities of its derivatives typically occur in the minima, which prohibits the use of standard optimization methods. In order to avoid the problem of non-smoothness, we use the pole-placement method in the vicinity of non-smooth minimum.

For simplicity, here we restrict ourselves to the case of the scalar input variable, $m = 1$ (the extension of the algorithm to the case of arbitrary m is straightforward). Then $\mathbf{b}$ and $\mathbf{k}$ represent the vectors of the dimension d, the same as the dimension of the state vector $\mathbf{x}$. First we formulate the main steps of our algorithm.

2.2.1 Main steps of the algorithm

Step 1. For a given input vector $\mathbf{b}$ choose some initial control vector $\mathbf{k}$. This choice can be rather arbitrary, however, for the examples presented below we consider the systems, which have been already stabilized by TDFC and take the value of $\mathbf{k}$ used in the literature (e.g. [32]) as an initial value.

Step 2. Scan the dependence of several FEs with the largest real parts on the feedback gain κ. To this end the DDE-BIFTOOL package [44] can be used. Find the minimal value of the real part $\Lambda_{\min}$ of the leading FE and the corresponding feedback gain $\kappa_{\min}$.

Step 3. By trial and error, select for some κ_1 (close to $\kappa_{\min}$) a set of FEs whose real parts should be shifted/held towards more negative values and compute the matrix of sensitivity.

Step 4. Shift/hold the real parts of the chosen FEs and perform simultaneously the Newton-Broyden iterations in order to minimize the error of the shifts. The control parameters are changed iteratively. When the Newton-Broyden method does not yield satisfactory corrections, stop shifting and go to Step 2. Repeat Steps 2−4 until the iterations of $\Lambda_{\min}$ do not reach the minimum. The control vector obtained at the last iteration is as an optimal vector.

The first two steps do not require additional explanation, while we describe the two last steps in more details below.

2.2.2 Computing the matrix of sensitivity

Our aim is to induce some small desired changes into real parts of several leading FEs by applying small carefully estimated perturbations to the components of the control vector. Such a manipulation with FEs can be accomplished by using the matrix of sensitivity. Suppose that at a fixed coupling strength κ_1 we selected N_{Fl} nontrivial FEs λ_i, $i = 1, \ldots, N_{Fl}$ with the largest real parts, which we intend to control. If there are complex conjugate pairs of exponents we include in this set only one exponent for each pair, since their real parts coincide.

We define the matrix of sensitivity $\mathbf{s}$ as a matrix of derivatives of the selected set of FEs with respect to the components of the control vector, $s_{ij} = \partial\lambda_i/\partial k_j$ with $i = 1, \ldots, N_{Fl}$ and $j = 1, \ldots, d$. To estimate this matrix let us rewrite the Floquet equation (4) for the set of selected FEs as:

$$\hat{L}_i \mathbf{u}_i(t) = \mathbf{0}, \tag{6}$$

where we introduced a linear operator

$$\hat{L}_i \equiv \frac{d}{dt} - \mathbf{A}(t) + \lambda_i \mathbf{I} + \kappa \mathbf{K}\left[1 - e^{-\lambda_i \tau}\right]. \tag{7}$$

In what follows we will need an equation adjoint to equation (6). First let us define an inner product of two τ-periodic vector functions $\mathbf{v}(t)$ and $\mathbf{u}(t)$ as:

$$\langle \mathbf{v}|\mathbf{u}\rangle \equiv \int_0^\tau dt \mathbf{v}^\dagger(t)\mathbf{u}(t), \tag{8}$$

where $\mathbf{v}^\dagger$ is the conjugate transpose of the column vector $\mathbf{v}$. Then we introduce a standard definition for an adjoint operator $\hat{L}^\dagger$:

$$\langle \hat{L}^\dagger \mathbf{v}|\mathbf{u}\rangle = \langle \mathbf{v}|\hat{L}\mathbf{u}\rangle. \tag{9}$$

From equations (7)–(9) it follows an expression for the adjoint operator

$$\hat{L}_i^\dagger \equiv -\frac{d}{dt} - \mathbf{A}^T(t) + \lambda_i^* \mathbf{I} + \kappa \mathbf{K}^T[1 - e^{-\lambda_i^* \tau}] \tag{10}$$

and the equation adjoint to equation (6) reads:

$$\hat{L}_i^\dagger \mathbf{v}_i(t) = \mathbf{0}, \tag{11}$$

where $\mathbf{v}_i(t)$ as well as $\mathbf{u}_i(t)$ is the τ-periodic function.

Now we differentiate equation (6) with respect to all components of the control vector:

$$\frac{\partial \hat{L}_i}{\partial k_j}\mathbf{u}_i(t) + \hat{L}_i \frac{\partial \mathbf{u}_i(t)}{\partial k_j} = \mathbf{0}. \tag{12}$$

Then we multiply this equation from the left by $\mathbf{v}_i^\dagger(t)$ and integrate with respect to time over the period τ:

$$\left\langle \mathbf{v}_i \middle| \frac{\partial \hat{L}_i}{\partial k_j}\mathbf{u}_i \right\rangle + \left\langle \mathbf{v}_i \middle| \hat{L}_i \frac{\partial \mathbf{u}_i}{\partial k_j} \right\rangle = 0. \tag{13}$$

Due to the property (9) and equation (11) the last term in the left hand side of this equation vanishes:

$$\left\langle \mathbf{v}_i \middle| \hat{L}_i \frac{\partial \mathbf{u}_i}{\partial k_j} \right\rangle = \left\langle \hat{L}_i^\dagger \mathbf{v}_i \middle| \frac{\partial \mathbf{u}_i}{\partial k_j} \right\rangle = 0. \tag{14}$$

We are thus left with the equation

$$\left\langle \mathbf{v}_i \middle| \frac{\partial \hat{L}_i}{\partial k_j}\mathbf{u}_i \right\rangle = 0 \tag{15}$$

from which using (7) and (5) the values for the matrix of sensitivity can be extracted

$$s_{ij} = \left(\mu_i^{-1} - 1\right) \frac{\langle \mathbf{v}_i | \kappa \left(\mathbf{b}\mathbf{e}_j^T\right) | \mathbf{u}_i \rangle}{\langle \mathbf{v}_i | \mathbf{I} + \tau \mu_i^{-1} \kappa (\mathbf{b}\mathbf{k}^T) | \mathbf{u}_i \rangle}. \quad (16)$$

Here $\mathbf{e}_j$ is a d-dimensional column vector in which all the elements are zeros, except of the jth element, which is equal to unity. $\mu_i \equiv e^{\lambda_i \tau}$ is the notation for the Floquet multiplier. We see that the computation of the matrix of sensitivity requires a knowledge of the FEs λ_i and the Floquet functions of the direct $\mathbf{u}_i(t)$ and adjoint $\mathbf{v}_i(t)$ Floquet problems. Thus we need to solve two BVP problems defined by equations (6) and (11).

Alternatively, one may also estimate the matrix of sensitivity using the finite differences. Such an approach is less accurate but much simpler, since it does not require the knowledge of the Floquet functions $\mathbf{u}_i(t)$ and $\mathbf{v}_i(t)$. Suppose that (for a given κ) we know the dependence of the FEs λ_i on the control vector $\mathbf{k}$, $\lambda_i = \lambda_i(\mathbf{k})$. Then replacing the derivative by a finite difference the matrix of sensitivity can be estimated as:

$$s_{ij} = \left[\lambda_i \left(\mathbf{k} + \mathbf{e}_j \varepsilon\right) - \lambda_i \left(\mathbf{k} - \mathbf{e}_j \varepsilon\right)\right] / 2\varepsilon, \quad (17)$$

where ε is a sufficiently small positive parameter.

2.2.3 Shifting the real parts of Floquet exponents

Assume that we are going to control the real parts $\Lambda_i \equiv \mathrm{Re}\lambda_i$, $i = 1, \ldots, N_{Fl}$ of $N_{Fl} \leq d$ selected nontrivial FEs. Let us denote the desired (small) displacement of the real parts of controlled FEs by a vector $\Delta\mathbf{\Lambda} = [\Delta\Lambda_1 \ldots \Delta\Lambda_{N_{Fl}}]^T$. Then one can compute a change of $\Delta\mathbf{k}$ for the given $\mathbf{k}$ such that

$$\mathbf{S}\Delta\mathbf{k} = \Delta\mathbf{\Lambda}, \quad (18)$$

where $\mathbf{S} \equiv \mathrm{Re}(\mathbf{s})$ is the real part of the matrix of sensitivity (16). For $N_{Fl} < d$ the number of equations in system (18) is less than the number of unknowns and this leads to infinitely many solutions. One possibility to determine a unique solution consists of controlling the N_{Fl} FEs using only N_{Fl} selected components of $\mathbf{k}$ and taking $d - N_{Fl}$ components of $\Delta\mathbf{k}$ equal to zero. Another possibility, consists of taking the solution with $||\Delta\mathbf{k}||$ minimal (see Ref. [36] for motivation). Then $\Delta\mathbf{k}$ is obtained via pseudoinverse

$$\Delta\mathbf{k} = \mathbf{S}^+ \Delta\mathbf{\Lambda}, \quad (19)$$

where $\mathbf{S}^+$ is a pseudoinverse of the matrix $\mathbf{S}$. Technically, the pseudoinversion may be performed by using the singular value decomposition.

With the new feedback gain $\mathbf{k} + \Delta\mathbf{k}$, the displacement of the controlled FEs will generically not be equal to $\Delta\mathbf{\Lambda}$, since equation (18) is based on linearization, and some correction is needed. The exact $\Delta\mathbf{k}$ satisfies the nonlinear system

$$\mathbf{F}(\Delta\mathbf{k}) \equiv \mathbf{\Lambda}(\mathbf{k} + \Delta\mathbf{k}) - \mathbf{\Lambda}(\mathbf{k}) - \Delta\mathbf{\Lambda} = 0. \quad (20)$$

This system could be solved by the standard Newton-iterative method. In this case we would need to estimate the Jacobian of the function $\mathbf{F}$ (which coincides with the matrix of sensitivity $\mathbf{S}$) at each step of the iteration. A more convenient approach is to use the Newton-Broyden algorithm [45] that incorporates an iterative equation for the Jacobian. For our system (20) this algorithm reads:

$$\Delta\mathbf{k}_{n+1} = \Delta\mathbf{k}_n + \delta\mathbf{k}_n, \quad (21)$$

$$\mathbf{S}_{n+1} = \mathbf{S}_n + \frac{(\mathbf{F}_{n+1} - \mathbf{F}_n - \mathbf{S}_n \delta\mathbf{k}_n)\delta\mathbf{k}_n^T}{\delta\mathbf{k}_n^T \delta\mathbf{k}_n}, \quad (22)$$

where $\delta\mathbf{k}_n = -\mathbf{S}_n^+ \mathbf{F}_n$ and $\mathbf{F}_n = \mathbf{F}(\Delta\mathbf{k}_n)$. The iterations start with the initial value of the Jacobian $\mathbf{S}_0$ obtained from equation (16) and the initial $\Delta\mathbf{k}_0$ defined from equation (19). Only a few Newton-Broyden iterations on equation (20) are needed when $\Delta\mathbf{k}$ is sufficiently small.

Here we should make some remarks about the third step. Although this step relies on intuition, it is not hard to apply it practically. The main criteria for selection (by trial and error) of the set of the FEs and the feedback gain are as follows. The Newton-Broyden iterations should converge fast and the number of deepened FEs should be as large as possible. If the shifting of all selected roots into negative direction does not yield fast convergence of Newton-Broyden iterations, one may try to shift one of them, and to keep fixed others.

2.3 Numerical demonstration

Now we demonstrate our algorithm for the Lorenz system [42] described by equation (1) with the state vector $\mathbf{x} = [x_1, x_2, x_3]^T$ and the vector field

$$\mathbf{f}(\mathbf{x}) = [\sigma(x_2 - x_1), rx_1 - x_2 - x_1 x_3, x_1 x_2 - b x_3]^T \quad (23)$$

at the standard values of the parameters $\sigma = 10$, $r = 28$, and $b = 8/3$. For a long time, this system has been considered as a classical example of a system inaccessible by the standard TDFC method due to the odd number limitation. The form of the control matrix that stabilizes the period-one UPO of this system has been found only recently [32]. However, the problem of optimization of the control matrix has not been considered in reference [32]. Here we solve this problem.

The symmetric period-one UPO with the period $\tau \approx 1.55865$ and the single unstable FM $\mu \approx 4.713$ has been stabilized in reference [32] with the feedback law defined by equation (1) using the input vector $\mathbf{b} = [0, 1, 0]^T$ and the control vector $\mathbf{k} = [-1, 0, 0.5]^T$. The real parts $\Lambda = \mathrm{Re}(\lambda)$ of several leading FEs as functions of the coupling strength κ for these specific values of $\mathbf{b}$ and $\mathbf{k}$ are shown in Figure 1a. We see that the best choice for the coupling strength is $\kappa_{\min} \approx 0.865$, where the nontrivial leading FE (with the maximal real part) $\Lambda_{\sup}(\kappa)$ reaches the minimum $\Lambda_{\min} \equiv \Lambda_{\sup}(\kappa_{\min}) \approx -0.4009$. In Figure 1b, we show a local enlargement of Figure 1a around the point $(\kappa_{\min}, \Lambda_{\min})$. Our aim is to modify the components of the control vector $\mathbf{k}$ in such a way as to minimize the value

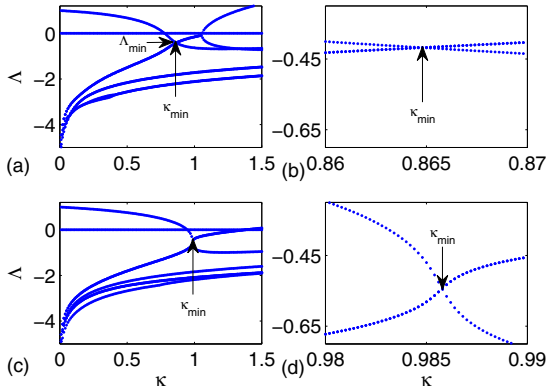

Fig. 1. Real parts of FEs vs. feedback gain for the period-one UPO of the Lorenz system. (a) The control vector $\mathbf{k} = [-1, 0, 0.5]^T$ is taken from reference [32]. (c) The value of the control vector $\mathbf{k} = \mathbf{k}_{\rm opt} = [-0.92972, 0.14974, 0.39354]^T$ is optimized by our algorithm. (b) and (d) The enlarged parts of diagrams (a) and (c), respectively, around the point $(\kappa_{\rm min}, \Lambda_{\rm min})$. At the start, the minimum of the leading FE is $\Lambda_{min} \approx -0.4009$ for $\kappa_{\rm min} \approx 0.865$, and after the optimization, it is $\Lambda_{\rm min} \approx -0.5426$ for $\kappa_{\rm min} \approx 0.986$.

Table 1. Shifting the FEs in the Lorenz system.

No.	$\mathbf{k}^T$	κ_1	FEs	$\Delta\mathbf{\Lambda}^T/10^{-3}$	$N_{\rm it}$
1	$[-1, 0, 0.5]$	0.95	[II, IV]	$[-1, 0]$	10
2	$[-0.99773, 0.00459, 0.49680]$	0.95	[II, IV]	$[0, -1]$	10
3	$[-0.99709, 0.00651, 0.49456]$	0.95	[II, IV]	$[-1, -1]$	39
4	$[-0.98537, 0.03225, 0.47374]$	0.95	[II, IV]	$[0, -1]$	25
5	$[-0.98319, 0.03774, 0.46836]$	0.95	[II, IV]	$[-1, 0]$	30
6	$[-0.97639, 0.05141, 0.45899]$	0.95	[II, IV]	$[-1, 0]$	30
7	$[-0.96965, 0.06509, 0.44943]$	0.95	[II, IV]	$[-1, 0]$	30
8	$[-0.96288, 0.07896, 0.43955]$	0.97	[II, IV]	$[-1, 0]$	40
9	$[-0.95418, 0.09791, 0.42708]$	0.97	[II, IV]	$[-1, 0]$	20
10	$[-0.94973, 0.10772, 0.42049]$	1.0	[II, IV]	$[-1, 0]$	40
11	$[-0.94060, 0.12674, 0.40851]$	1.0	[II, IV]	$[-1, 0]$	45

$\Lambda_{\rm min}$. We have achieved this goal by applying the above algorithm.

All the performed shifts of the FEs are recorded in Table 1. The first column gives the number of the stride inside of which $N_{\rm it}$ iterations are performed (the last column). The second column shows the control vector at which the current stride is started. The third column shows the value of the feedback gain where we apply the shifts. The numbers of FEs selected for control are presented in the fourth column. The FEs are enumerated by Roman numerals in descending order of their real parts. The fifth column shows the desired shifts of the selected FEs at each iteration. The evaluations of the matrix of sensitivity using equation (16) were performed only at the start of strides No. 1, No. 8 and No. 10, at which the value of feedback gain κ_1 was changed. Otherwise, the matrix of sensitivity from the previous stride was taken as an initial guess for the Newton-Broyden iterations.

For graphical illustration of intermediate results, in Figures 2 and 3 we show the changes of the control vector

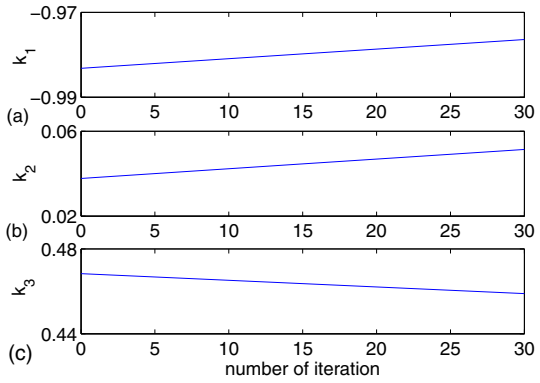

Fig. 2. Iterations of the control vector $\mathbf{k}$ at the stride No. 5 in Table 1.

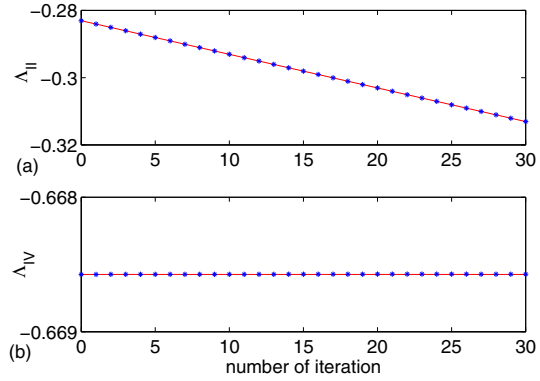

Fig. 3. Iterations of the selected FEs at the stride No. 5 in Table 1. The crosses show the actual shifts of FEs, while the solid lines connect the start point with the desired end point.

and the selected FEs, respectively, resulting from the application of our algorithm at the stride No. 5 in Table 1. We see that the algorithm performs perfectly the desired linear decrease of the FE $\Lambda_{\rm II}$ and conservation of the FE $\Lambda_{\rm IV}$. Note that although the FE $\lambda_{\rm III}$ is not selected for control, its real part $\Lambda_{\rm III}$ moves during the iterations identically as $\Lambda_{\rm II}$, since these two exponents represent a complex conjugate pair.

The final results obtained with our algorithm after application of 11 strides listed in Table 1 are presented in Figures 1c and 1d. Figure 1c shows the global view of dependence Λ vs. κ, while Figure 1d represents an enlargement of this dependence around the point $(\kappa_{\rm min}, \Lambda_{\rm min})$. Comparing Figures 1b and 1d we see that our algorithm allowed us to notably reduce the minimum of the leading FE. The final $\Lambda_{\rm min} \approx -0.5426$ is attained with the control vector

$$\mathbf{k}_{\rm opt} = [-0.92972, 0.14974, 0.39354]^T \qquad (24)$$

that can be interpreted as an optimal control vector.

3 Extension to the ETDFC systems

Now we extend our algorithm for systems controlled by extended time-delayed feedback. As well as previously,

we consider the stabilization of an UPO in system described by equation (1) and restrict ourselves to the case of the scalar input variable $u_{\rm in}$. Then the extended time-delayed feedback controller can be presented in the form [4]:

$$u_{\rm in}(t) = -\kappa \mathbf{k}^T \left[\mathbf{x}(t) - (1-R)\mathbf{z}(t-\tau)\right], \quad (25)$$

$$\mathbf{z}(t) = \mathbf{x}(t) + R\mathbf{z}(t-\tau), \quad (26)$$

where R is a so-called memory parameter. The ETDFC law (25) and (26) is noninvasive as well as the original TDFC (2), since for any periodic function $\mathbf{x}(t) = \mathbf{x}(t-\tau)$ the input variable vanishes $u_{\rm in}(t) = 0$. For $R = 0$, the ETDFC reduces to the original TDFC, since in this case $\mathbf{z}(t) = \mathbf{x}(t)$ and equation (25) transforms to equation (2).

When optimizing the ETDFC, we can vary not only the control vector $\mathbf{k}$, but also the memory parameter R. Thus besides the matrix of sensitivity $s_{ij}^k = \partial\lambda_i/\partial k_j$ with respect to $\mathbf{k}$ we need also to estimate the vector of sensitivity $s_i^R = \partial\lambda_i/\partial R$ with respect to R. Straightforward application of the ideas presented in Section 2 leads to the following expressions for these derivatives:

$$s_{ij}^k = -H_0 \frac{\langle \mathbf{v}_i | \kappa(\mathbf{b}\mathbf{e}_j^T) | \mathbf{u}_i \rangle}{\langle \mathbf{v}_i | \mathbf{D} | \mathbf{u}_i \rangle}, \quad (27)$$

$$s_i^R = -\frac{H_0 \mu_i^{-1}}{(1 - R\mu_i^{-1})} \frac{\langle \mathbf{v}_i | \kappa(\mathbf{b}\mathbf{k}^T) | \mathbf{u}_i \rangle}{\langle \mathbf{v}_i | \mathbf{D} | \mathbf{u}_i \rangle}. \quad (28)$$

Here we have introduced the following notations:

$$\mathbf{D} = \mathbf{I} + H_1 \tau \kappa \left(\mathbf{b}\mathbf{k}^T\right), \quad (29)$$

$$H_0 = \left(1 - \mu_i^{-1}\right) / \left(1 - R\mu_i^{-1}\right), \quad (30)$$

$$H_1 = \mu_i^{-1}(1-R) / \left(1 - R\mu_i^{-1}\right)^2. \quad (31)$$

Note that the Floquet functions $\mathbf{u}_i$ and $\mathbf{v}_i$ are computed from the same BVP equations (6), (7) and (10), (11), with formal substitutions

$$\mathbf{K}\left(1 - e^{-\lambda_i \tau}\right) \to \mathbf{K}\left(1 - e^{-\lambda_i \tau}\right) / \left(1 - e^{-\lambda_i \tau} R\right)$$

and

$$\mathbf{K}^T\left(1 - e^{-\lambda_i^* \tau}\right) \to \mathbf{K}^T\left(1 - e^{-\lambda_i^* \tau}\right) / \left(1 - e^{-\lambda_i^* \tau} R\right),$$

respectively (see e.g. [46]). As is expected, for $R = 0$, equation (27) reduces to equation (16).

Defining a joined vector $\mathbf{p} = [\mathbf{k}^T, R]^T$ and a joined matrix of sensitivity $\mathbf{S} = [\mathbf{S}^k, \mathbf{S}^R]$, we can write for the shift of $\Delta\mathbf{p}$ the equation $\mathbf{S}\Delta\mathbf{p} = \Delta\mathbf{\Lambda}$, similar to equation (18) and determine this shift via pseudoinverse $\Delta\mathbf{p} = \mathbf{S}^+\Delta\mathbf{\Lambda}$, similar to equation (19). We thus apply the methodic described in Section 2.2.3 after formal substitution $\mathbf{k} \to \mathbf{p}$. The same is valid for equation (17) with a correction that $\mathbf{e}_j$ is now a $(d+1)$-dimensional vector.

Finally, we emphasize the problem of computation of FEs for the ETDC systems. Unfortunately, the standard DDE-BIFTOOL package does not apply for such systems due to the presence of the difference equation (26). To compute FEs for the ETDFC systems, we have developed a spectral element method, which is described in Appendix.

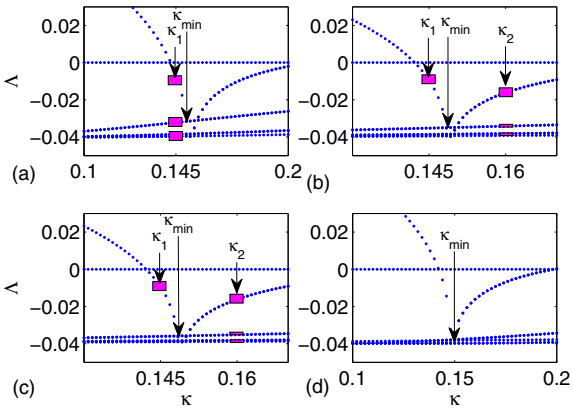

Fig. 4. Real parts of FEs vs. feedback gain for the period-four UPO of the Rössler system. (a) The initial control vector is $\mathbf{k} = [0, 1, 0]^T$ and the value of the memory parameter $R = 0.39$ is optimal for the given SISO scheme. (b)–(d) correspond to the end of iterations at the strides No. 1, No. 2 and No. 3 in Table 2. The rectangles in (a)–(c) show the selected FEs for the control at the beginning of strides No. 1, No. 2 and No. 3, respectively. In (a) the minimum of the leading FE is $\Lambda_{\rm min} \approx -0.03142$ (the start value) for $\kappa_{\rm min} \approx 0.15$, while in (d) it is $\Lambda_{\rm min} \approx -0.03766$ (the final value) for $\kappa_{\rm min} \approx 0.15$.

Numerical demonstration

As an illustrative example, we take the Rössler system [43] described by equation (1) with the state vector $\mathbf{x} = [x_1, x_2, x_3]^T$ and the vector field

$$\mathbf{f}[\mathbf{x}] = [-x_2 - x_3, x_1 + ax_2, b + x_3(x_1 - c)]^T \quad (32)$$

at the standard values of the parameters $a = 0.2$, $b = 0.2$, and $c = 5.7$. Here we consider a relatively complicated problem, the stabilization of the period-four UPO with the period $\tau \approx 23.50362$ and initial state vector $[x_1, x_2, x_3]^T \approx [-4.14784, 0.00781, 0.02042]^T$. This orbit (as many other high-periodic orbits) cannot be stabilized by the standard TDFC algorithm with any control matrix, but the stabilization can be achieved by the ETDFC. First we consider the ETDFC in a single-input-single-output (SISO) scheme with the input vector $\mathbf{b} = [0, 1, 0]^T$ and control vector $\mathbf{k} = [0, 1, 0]^T$. Using the ETDFC law (25) and (26), we scanned the feedback gain κ with various values of the memory parameter R, and found that for $R = 0.39$ the minimum of the leading FE $\Lambda_{\rm min} \approx -0.03142$ is the deepest (Fig. 4a).

Now we formulate our optimization problem. We assume that the input vector $\mathbf{b} = [0, 1, 0]^T$ is fixed, but the control vector and the memory parameter can be adjusted. Thus instead of SISO we consider a single-input-multiple-output (SIMO) scheme. Our aim is to shift the given minimum $\Lambda_{\rm min}$ as deep as possible by small iterative steps, applying necessary adjustments to the control parameters $(\mathbf{k}, R)$. In Figure 4a we see that for the initial values of the control parameters, below the minimum $\Lambda_{\rm min} \approx -0.03142$ attained at $\kappa_{\rm min} \approx 0.15$, there is a branch on the level of ≈ -0.03833. This branch represents a "bottom" that we are going to reach. Under this

Table 2. Shifting the FEs in the Rössler system.

No.	R	$\mathbf{k}^T$	κ_1	κ_2	FEs (κ_1)	FEs (κ_2)	$\Delta\mathbf{\Lambda}_1/10^{-3}$	$\Delta\mathbf{\Lambda}_2/10^{-3}$	$N_{\rm it}$
1	0.39	$[0, 1, 0]$	0.145	–	[II, III, V]	–	$[-1, -1, 0]$	–	3
2	0.38616	$[0.22479, 0.82894, -0.07047]$	0.145	0.16	[II]	[II, IV, VI]	[0]	$[0, -1, 0]$	1
3	0.39527	$[0.18714, 0.75439, -0.28662]$	0.145	0.16	[II]	[II, IV, VI]	[0]	$[0, -0.5, 0]$	4

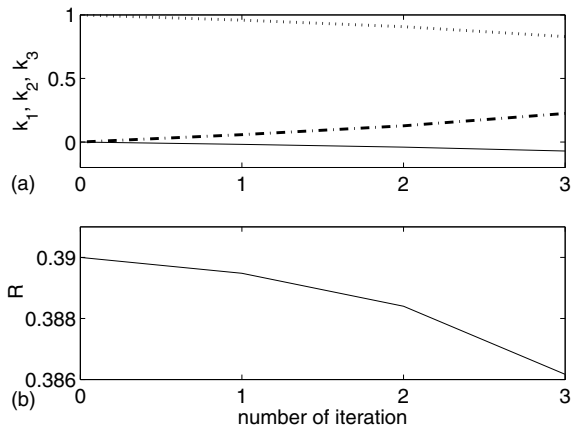

Fig. 5. Iterations of (a) the components k_1 (dash-dotted line), k_2 (dotted line) and k_3 (solid line) of the control vector $\mathbf{k}$ and (b) the memory parameter R for the stride No. 1 in Table 2.

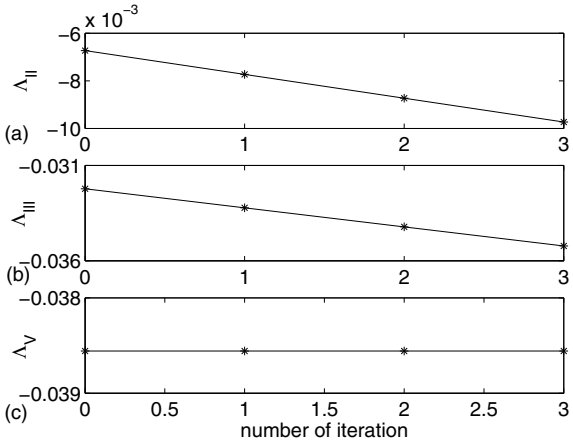

Fig. 6. Iterations of the selected FEs on the stride No. 1 in Table 2. The crosses show the actual shifts of FEs, while the solid lines connect the start point with the desired end point.

branch there are lots of other closely lying delay-induced branches. We will not touch them since their number is too large compared with the number of available control parameters. Next, in the global picture we see the two other important branches, one of them on the left hand side and another on the right hand side from $\kappa_{\rm min}$. If we intend to sink one of these branches, the other tends to rise, and vice versa. This happens because of the structure of the matrix of sensitivity. Hence, summing up, we have in general to move (or keep) four FEs, whereas there are available four adjustable control parameters $(\mathbf{k}, R)$.

Table 2 contains the information about the performed manipulations with the system parameters and its leading FEs. This table is composed in a similar way as Table 1. At the stride No. 1 we control the real parts of three exponents $\Lambda_{\rm II}$, $\Lambda_{\rm III}$ and $\Lambda_{\rm V}$ for the value of the coupling strength $\kappa_1 = 0.145$. In Figure 4a they are marked by rectangles. The exponent $\lambda_{\rm IV}$ represents a complex conjugate pair of $\lambda_{\rm III}$ and it is not selected for control. We perform three iterations at which we move down the exponents $\Lambda_{\rm II}$ and $\Lambda_{\rm III}$ by value $\Delta\Lambda = 10^{-3}$ at each iteration and keep the exponent $\Lambda_{\rm V}$ unchanged. The variation of the control parameters $(\mathbf{k}, R)$ and the corresponding changes of the FEs at the stride No. 1 are shown in Figures 5 and 6, respectively. The final dependence of the FEs on the coupling strength at the end of stride No. 1 is shown in Figure 4b.

At the next two strides No. 2 and No. 3 we proceed in a slightly different manner. Here we control the FEs at two different values κ_1 and κ_2 of the feedback gain simultaneously. The values $\kappa_1 = 0.145$ and $\kappa_2 = 0.16$ are taken from different sides of $\kappa_{\rm min}$, so that $\kappa_1 < \kappa_{\rm min} < \kappa_2$. The selected FEs at κ_1 and κ_2 are presented in 6th and 7th columns of Table 2 and displayed by rectangles in Figure 4b, while the corresponding shifts $\Delta\mathbf{\Lambda}_1$ and $\Delta\mathbf{\Lambda}_2$ are given in columns 8 and 9, respectively. The generalization of our algorithm for simultaneous control of the FEs at two different values of the feedback gain is straightforward. Here we define the vector $\mathbf{\Lambda}$ as a vector composed of two sub-vectors $\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ evaluated at different feedback gains κ_1 and κ_2, respectively. Accordingly, the matrix $\mathbf{S}$ is also composed of two sub-matrices defining the sensitivity of selected FEs at different values of the feedback gain. Then the whole mathematics remains the same as described above. The dependencies of the FEs on the feedback gain at the end of strides No. 2 and No. 3 are shown in Figures 4c and 4d, respectively. To the end of the stride No. 3 we managed to decrease the minimum of the leading FE to the value $\Lambda_{\rm min} \approx -0.03766$. This value has been attained at $\kappa_{\rm min} \approx 0.15$ with the following values of the control vector and memory parameter:

$$\mathbf{k}_{\rm opt} = [0.05674, 0.75475, -0.45264]^T, \quad R_{\rm opt} = 0.39298.$$

These parameters can be considered as the optimal parameters for the given ETDFC SIMO scheme.

4 Conclusions

In this paper, we have elaborated an algorithm for the optimization of the control matrix in time-delayed feedback controlled system in order to minimize the time at which the controlled system approaches the stabilized periodic orbit. To this end the pole placement method for linear time-invariant systems with time-delay [36] has been

adapted to time-delay systems with the periodic time-depended coefficients. The algorithm exploits the property of the continuous dependence of the Floquet exponents on the control parameters. The pole placement problem in time-delayed systems is nontrivial, since the number of Floquet exponents is infinite, while the number of the control parameters is finite. Nevertheless, several leading Floquet exponents can be controlled in desired fashion by applying carefully evaluated variations to the available control parameters. In order to evaluate these variations we use the matrix of sensitivity, which represents the derivatives of the Floquet exponents with respect to the control parameters. Appealing to the Floquet theory, we have derived an analytical expression for this matrix.

The proposed algorithm is demonstrated numerically for the Lorenz system, which until recently has been considered as a system inaccessible for the time-delayed feedback control due to the odd number limitation. We have also presented an extension of our algorithm to the case of the extended time-delayed feedback control and demonstrated this approach for the Rössler system.

Here we have considered only one specific optimization problem in which we required the fastest decay of deviations from the stabilized orbit, however, our algorithm can be easily extended to other optimization problems. For example, the algorithm can be straightforwardly adapted to the problem of finding optimal control parameters that maximize the interval of stability of the coupling strength.

This research was funded by the European Social Fund under the Global Grant measure (Grant No. VP1-3.1-ŠMM-07-K-01-025).

Appendix: Spectral element method for extended time-delayed feedback controlled systems

Here we describe the spectral element method for computation of FEs in the ETDFC systems. The DDE-BIFTOOL package [44] does not apply to this case, since it cannot operate with the difference equations. In the ETDFC law (25) and (26), such an equation is equation (26). Formally, the solution of this equation can be found by iterations and presented in the form of the infinite sum:

$$\mathbf{z}(t) = \sum_{n=0}^{\infty} R^n \mathbf{x}(t - n\tau). \tag{A.1}$$

In reference [47] (see also the references therein), the spectral element method for the ETDFC systems has been proposed, in which an infinite sum (A.1) was truncated, i.e., the solution of the difference equation (26) was approximated by a finite sum. Such an approximation may be costly for computation, since it may require a large number of terms in the sum to guarantee a desired accuracy. Especially for $|R|$ close to 1 the convergence of the sum is slow and such an approximation may lead to large errors. Here we modify the spectral element method [47] in such a way that we do not use any truncated solutions for equation (26), instead we incorporate this difference equation into the algorithm explicitly.

In the general form, the linearized ETDFC system reads:

$$\dot{\widetilde{\mathbf{x}}}(t) = \widetilde{\mathbf{A}}(t)\widetilde{\mathbf{x}}(t) + (1-R)\widetilde{\mathbf{K}}\widetilde{\mathbf{z}}(t-\tau) \tag{A.2}$$

$$\widetilde{\mathbf{z}}(t) = \widetilde{\mathbf{x}}(t) + R\widetilde{\mathbf{z}}(t-\tau). \tag{A.3}$$

Here $\widetilde{\mathbf{x}}(t) = \mathbf{x}(t) - \xi(t)$ is the deviation of the solution from the UPO, and the original Jacobian is shifted: $\widetilde{\mathbf{A}}(t) = \mathbf{A}(t) - \kappa\mathbf{K}$. The feedback gain is here absorbed in the control matrix: $\widetilde{\mathbf{K}} = \kappa\mathbf{K}$, and $\widetilde{\mathbf{z}}(t) = \mathbf{z}(t) - \xi(t)/(1-R)$ is the deviation of the auxiliary variable from its periodic state.

First we expand the vectors $\widetilde{\mathbf{x}}(t)$ and $\widetilde{\mathbf{z}}(t)$ by the Lagrange polynomials ϕ_i:

$$\widetilde{\mathbf{x}}_j(t) = \sum_{i=1}^{n+1} \widetilde{\mathbf{x}}_{ji}\phi_i(\eta), \tag{A.4}$$

$$\widetilde{\mathbf{z}}_j(t) = \sum_{i=1}^{n+1} \widetilde{\mathbf{z}}_{ji}\phi_i(\eta). \tag{A.5}$$

The corresponding delayed expressions read:

$$\widetilde{\mathbf{x}}_j(t-\tau) = \sum_{i=1}^{n+1} \widetilde{\mathbf{x}}_{ji-n}\phi_i(\eta), \tag{A.6}$$

$$\widetilde{\mathbf{z}}_j(t-\tau) = \sum_{i=1}^{n+1} \widetilde{\mathbf{z}}_{ji-n}\phi_i(\eta). \tag{A.7}$$

Here η is the local time normalized from 0 to 1 for the element j. A total of $n+1$ interpolation nodes are used for each element. First, we consider the case with one element. For the interpolation nodes we have chosen the Legendre-Gauss-Lobatto (LGL) points. They are the roots of the polynomial $(1-u^2)L_n'(u)$ where $L_n(u)$ is the nth order Legendre function, $L_n'(u)$ is its derivative with respect to u, and u is defined in the range $[-1, 1]$. Therefore, the LGL nodes must be shifted to the range $[0, 1]$ in order to be compatible with equations (A.4) and (A.5). The trial functions ϕ_i can be found from the barycentric Lagrange formula (see, e.g., Eqs. (11) and (12) in [47]). The trial functions have the useful property

$$\phi_i(t_k) = \delta_{i,k} \tag{A.8}$$

for $i, k = 1, \ldots, n+1$. The derivatives of the trial functions can also be calculated using the barycentric formula. For a matrix $\mathbf{D}$ with elements $D_{ki} = \phi_i'(t_k)$, the derivative of a vector states $\mathbf{w}$ on a mesh of LGL nodes is given by $\mathbf{w}' = \mathbf{D}\mathbf{w}$.

In view of the relations (A.4)–(A.7), the recurrent expression (A.3) now can be written as:

$$\widetilde{\mathbf{z}}_{ji} = \widetilde{\mathbf{x}}_{ji} + R\widetilde{\mathbf{z}}_{ji-n}. \tag{A.9}$$

After inserting expansions (A.4)–(A.7) into original equation (A.2), and projecting this equation onto the finite basis of the Legendre polynomials $\psi_p(\eta)$, we arrive at:

$$\int_0^1 d\eta\psi_p(\eta)\Big[\widetilde{\mathbf{x}}_{ji}\phi_i'(\eta)\frac{1}{t_j}-\widetilde{\mathbf{A}}(t_\eta)\widetilde{\mathbf{x}}_{ji} - (1-R)\widetilde{\mathbf{K}}\widetilde{\mathbf{z}}_{ji-n}(t_\eta)\phi_i(\eta)\Big]=\mathbf{0}. \quad \text{(A.10)}$$

Here p is the order of the polynomials, η is the normalized time at each element, and $t_\eta=(\eta+j-1)t_j$. $t_j=\tau/E$ is the duration of the jth element (now $E=1$ is the number of elements). The speed of integration is increased by using the quadrature weights instead of the symbolic integration:

$$\int_0^1 d\eta f(\eta)\approx\sum_{k=1}^{n+1}w_k f(\eta_k). \quad \text{(A.11)}$$

Here w_k is the quadrature weight and η_k is the localized time at the node k. For a grid of LGL points, the quadrature weights are given by:

$$w_k=\begin{cases}\frac{2}{n(n+1)}, & k=1,n+1\\ \frac{2}{n(n+1)(L_n(\eta_k))^2}, & \text{otherwise.}\end{cases} \quad \text{(A.12)}$$

We thus have to compute the following discrete expressions:

$$\sum_{k=1}^{n+1}w_k\psi_p(\eta_k)\widetilde{\mathbf{x}}_{ji}\phi_i'(\eta_k)\frac{1}{t_j}=\widetilde{\mathbf{A}}(t_\eta)\widetilde{\mathbf{x}}_{ji}\psi_p(\eta_i)w_i + (1-R)\widetilde{\mathbf{K}}\widetilde{\mathbf{z}}_{ji-n}w_i\psi_p(\eta_i). \quad \text{(A.13)}$$

Denoting

$$\mathbf{N}_{ji}^p=\sum_{k=1}^{n+1}w_k\psi_p(\eta_k)\phi_i'(\eta_k)\,\mathbf{I}\frac{1}{t_j}-\widetilde{\mathbf{A}}(t_\eta)\psi_p(\eta_i)w_i \quad \text{(A.14)}$$

and

$$\mathbf{P}_{ji-n}^p=(1-R)\widetilde{\mathbf{K}}w_i\psi_p(\eta_i), \quad \text{(A.15)}$$

we get the matrix equations

$$\sum_{i=1}^{n+1}\mathbf{N}_{ji}^p\widetilde{\mathbf{x}}_{ji}=\sum_{i=1}^{n+1}\mathbf{P}_{ji-n}^p\widetilde{\mathbf{z}}_{ji-n} \quad \text{(A.16)}$$

$$\widetilde{\mathbf{z}}_{ji}=\widetilde{\mathbf{x}}_{ji}+R\widetilde{\mathbf{z}}_{ji-n}. \quad \text{(A.17)}$$

In the matrix notations we have

$$\mathbf{H}\widetilde{\mathbf{x}}_m=\mathbf{G}\widetilde{\mathbf{z}}_{m-1} \quad \text{(A.18)}$$

$$\widetilde{\mathbf{z}}_m=\widetilde{\mathbf{x}}_m+R\widetilde{\mathbf{z}}_{m-1}. \quad \text{(A.19)}$$

Here $\widetilde{\mathbf{x}}_m$ is the column vector of $[\widetilde{\mathbf{x}}_{ji}^T,\widetilde{\mathbf{x}}_{ji+1}^T,\ldots,\widetilde{\mathbf{x}}_{ji+n}^T]^T$, and $\widetilde{\mathbf{x}}_{m-1}$ is the column vector of $[\widetilde{\mathbf{x}}_{ji-n}^T,\widetilde{\mathbf{x}}_{ji-n+1}^T,\ldots,\widetilde{\mathbf{x}}_{ji}^T]^T$ (the same happens with indices of column vectors $\widetilde{\mathbf{z}}_m$ and $\widetilde{\mathbf{z}}_{m-1}$, respectively).

The matrices $\mathbf{H}$ and $\mathbf{G}$ are given by:

$$\mathbf{H}=\begin{bmatrix}\mathbf{I} & \mathbf{0} & \ldots & \mathbf{0}\\ \mathbf{N}_{ji}^1 & \mathbf{N}_{ji+1}^1 & \ldots & \mathbf{N}_{ji+n}^1\\ \ldots & \ldots & \ldots & \ldots\\ \mathbf{N}_{ji}^n & \mathbf{N}_{ji+1}^n & \ldots & \mathbf{N}_{ji+n}^n\end{bmatrix}$$

$$\mathbf{G}=\begin{bmatrix}-R\mathbf{I} & \mathbf{0} & \ldots & \mathbf{I}\\ \mathbf{P}_{ji-n}^1 & \mathbf{P}_{ji-n+1}^1 & \ldots & \mathbf{P}_{ji}^1\\ \ldots & \ldots & \ldots & \ldots\\ \mathbf{P}_{ji-n}^n & \mathbf{P}_{ji-n+1}^n & \ldots & \mathbf{P}_{ji}^n\end{bmatrix}. \quad \text{(A.20)}$$

The upper rows in $\mathbf{H}$ and $\mathbf{G}$ insure the validity of recurrent relation (A.17). In these rows, the identity matrices $\mathbf{I}$ are of $d\times d$ dimensions.

In order to find the FMs of the periodic solution of system (A.2), (A.3), we need to compute the matrices $\mathbf{H}$ and $\mathbf{G}$, and to solve the eigenvalues problem for the matrix of monodromy:

$$\mathbf{U}=\mathbf{H}^{-1}\mathbf{G}+R\mathbf{I}, \quad \text{(A.21)}$$

resulting from equation

$$\mathbf{H}\widetilde{\mathbf{z}}_m=(\mathbf{G}+R\mathbf{H})\widetilde{\mathbf{z}}_{m-1}. \quad \text{(A.22)}$$

This equation is deduced by multiplying equation (A.19) from left by matrix $\mathbf{H}$, and using expression (A.18). In equation (A.21) the unity matrix $\mathbf{I}$ has the same order as the matrix $\mathbf{H}^{-1}\mathbf{G}$.

However, such a situation takes the place only when there is a single spectral element, i.e. the whole period of integration is divided into $n+1$ nodes, and the dynamic system (with recurrent relation) is projected onto the Legendre polynomials. But the key idea of the spectral element method is to separate the period into E intervals (spectral elements), and each of them is in turn divided into $n+1$ nodes. In the following, we discuss the case of several spectral elements.

We thus have to deal with hyper-matrices containing several $\mathbf{G}_j$ and $\mathbf{H}_j$ matrices (their number is equal to E, i.e. $j=1,\ldots,E$) located almost diagonally, and the rest elements are zeros. These matrices differ from the single-element matrices as follows:

$$\mathbf{H}_j=\begin{bmatrix}\mathbf{N}_{ji}^1 & \mathbf{N}_{ji+1}^1 & \ldots & \mathbf{N}_{ji+n}^1\\ \ldots & \ldots & \ldots & \ldots\\ \mathbf{N}_{ji}^n & \mathbf{N}_{ji+1}^n & \ldots & \mathbf{N}_{ji+n}^n\end{bmatrix}$$

$$\mathbf{G}_j=\begin{bmatrix}\mathbf{P}_{ji-n}^1 & \mathbf{P}_{ji-n+1}^1 & \ldots & \mathbf{P}_{ji}^1\\ \ldots & \ldots & \ldots & \ldots\\ \mathbf{P}_{ji-n}^n & \mathbf{P}_{ji-n+1}^n & \ldots & \mathbf{P}_{ji}^n\end{bmatrix}. \quad \text{(A.23)}$$

They thus have lost their "hats" compared with original matrices (A.20). As an example, the hyper-matrices $\mathbf{H}$ and $\mathbf{G}$ for the case of $E=4$ are shown in Figure 7.

Note that the characteristic top is retained and became longer ($\mathbf{I}$ matrix in the left upper corner of the hyper-$\mathbf{H}$ matrix, and $-R\mathbf{I}$ on the left and $\mathbf{I}$ on the right in the upper

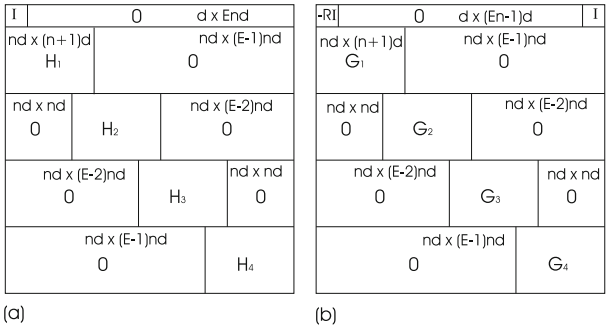

Fig. 7. Hyper-matrices **H** (a) and **G** (b) for $E = 4$. In the boxes the sizes of matrices are also given. The matrices **I** and $-R\mathbf{I}$ in the upper corners are of $d \times d$ dimensions.

rows of the hyper-**G** matrix). These upper rows insure that the recurrent relation, analogous to (A.17), is satisfied.

We thus need again to solve the eigenvalue problem for the matrix of monodromy (A.21), which results from new equation:

$$\mathbf{H}\widetilde{\mathbf{Z}}_m = (\mathbf{G} + R\mathbf{H})\widetilde{\mathbf{Z}}_{m-1}. \tag{A.24}$$

The column-vector $\widetilde{\mathbf{X}}_m$ (corresponding to the new column-vector $\widetilde{\mathbf{Z}}_m$) is composed as a concatenation of the column vectors corresponding to the each element:

$$\widetilde{\mathbf{X}}_m = \left[\widetilde{\mathbf{x}}_1^T, \widetilde{\mathbf{x}}_2^T, \ldots, \widetilde{\mathbf{x}}_E^T\right]^T. \tag{A.25}$$

The vectors $\widetilde{\mathbf{x}}_j^T$ are composed in the same way as for the single element with exception that the $(n+1)$st node of the jth element coincides with the 1st node of the $(j+1)$st element. This does not hold only for the $(n+1)$st node of the Eth element. One of the coinciding nodes in the vectors $\widetilde{\mathbf{x}}_j^T$ has always to be omitted. In the column-vector $\widetilde{\mathbf{X}}_{m-1}$, all the indices for nodes are shifted $(i \to i-n)$ compared wit those of $\widetilde{\mathbf{X}}_m$. The column-vectors $\widetilde{\mathbf{Z}}_m$ and $\widetilde{\mathbf{Z}}_{m-1}$ are composed by the same way as the vectors $\widetilde{\mathbf{X}}_m$ and $\widetilde{\mathbf{X}}_{m-1}$, respectively.

Summing up, we need to compute the hyper-matrices **H** and **G**, find the matrix of monodromy **U** from equation (A.21), and compute its eigenvalues which are the Floquet multipliers of the linearized ETDFC system (A.2) and (A.3).

References

1. E. Schöll, H.G. Schuster, *Handbook of Chaos Control* (Wiley-VCH, Weinheim, 2008)
2. K. Pyragas, Phys. Lett. A **170**, 421 (1992)
3. K. Pyragas, Philos. Trans. R. Soc. London, Ser. A **364**, 2309 (2006)
4. J.E.S. Socolar, D.W. Sukow, D.J. Gauthier, Phys. Rev. E **50**, 3245 (1994)
5. K. Pyragas, Phys. Lett. A **206**, 323 (1995)
6. T. Jüngling, A. Gjurchinovski, V. Urumov, Phys. Rev. E **86**, 046213 (2012)
7. A. Gjurchinovski, T. Jüngling, V. Urumov, E. Schöll, Phys. Rev. E **88**, 032912 (2013)
8. K. Höhne, H. Shirahama, C.U. Choe, H. Benner, K. Pyragas, W. Just, Phys. Rev. Lett. **98**, 214102 (2007)
9. A. Tamaševičius, G. Mykolaitis, V. Pyragas, K. Pyragas, Phys. Rev. E **76**, 026203 (2007)
10. K. Pyragas, V. Pyragas, Phys. Rev. E **80**, 067201 (2009)
11. V. Pyragas, K. Pyragas, Phys. Lett. A **375**, 3866 (2011)
12. V. Pyragas, K. Pyragas, Eur. Phys. J. B **86**, 306 (2013)
13. K. Yamasue, K. Kobayashib, H. Yamada, K. Matsushige, T. Hikihara, Phys. Lett. A **373**, 3140 (2009)
14. J. Sieber, A. Gonzalez-Buelga, S.A. Neild, D.J. Wagg, B. Krauskopf, Phys. Rev. Lett. **100**, 244101 (2008)
15. D.A.W. Barton, J. Sieber, Phys. Rev. E **87**, 052916 (2013)
16. A. Pimenov, A.G. Vladimirov, S.V. Gurevich, K. Panajotov, G. Huyet, M. Tlidi, Phys. Rev. A **88**, 053830 (2013)
17. M. Iñarrea, V. Lanchares, A.I. Pascual, J.P. Salas, Acta Astronautica **96**, 280 (2014)
18. K. Konishi, Y. Sugitani, N. Hara, Phys. Rev. E **89**, 022906 (2014)
19. S. Shao, K. Masri, M. Younis, Nonlinear Dyn. **74**, 257 (2013)
20. Y. Jin, H. Hu, Commun. Nonlinear. Sci. Numer. Simul. **18**, 1027 (2013)
21. M. Xiao, W.X. Zheng, J. Cao, Neural Netw. **44**, 132 (2013)
22. D.A. Strehober, E. Schöll, S.H.L. Klapp, Phys. Rev. E **88**, 062509 (2013)
23. J.B. Gonpe Tafo, L. Nana, T.C. Kofane, Phys. Rev. E **88**, 032911 (2013)
24. H. Nakajima, Phys. Lett. A **232**, 207 (1997)
25. H. Nakajima, Y. Ueda, Phys. Rev. E **58**, 1757 (1998)
26. K. Pyragas, Phys. Rev. Lett. **86**, 2265 (2001)
27. K. Pyragas, V. Pyragas, H. Benner, Phys. Rev. E **70**, 056222 (2004)
28. V. Pyragas, K. Pyragas, Phys. Rev. E **73**, 036215 (2006)
29. B. Fiedler, V. Flunkert, M. Georgi, P. Hövel, E. Schöll, Phys. Rev. Lett. **98**, 114101 (2007)
30. E.W. Hooton, A. Amann, Phys. Rev. Lett. **109**, 154101 (2012)
31. A. Amann, E.W. Hooton, Phil. Trans. R. Soc. A **371**, 20120463 (2013)
32. K. Pyragas, V. Novičenko, Phys. Rev. E **88**, 012903 (2013)
33. S. Steingrube, M. Timme, F. Worgotter, P. Manoonpong, Nat. Phys. **6**, 224 (2010)
34. R.C. Hinz, P. Hövel, E. Schöll, Chaos **21**, 023114 (2011)
35. J. Ackermann, Siam J. Optim. **7**, 297 (1972)
36. W. Michiels, K. Engelgorghs, P. Vansenant, D. Roose, Automatica **38**, 747 (2002)
37. J. Burke, A. Lewis, M. Overton, SIAM J. Optim. **15**, 751 (2005)
38. J. Vanbiervliet, K. Verheyden, W. Michels, S. Vandewalle, ESAIM: COCV **14**, 478 (2008)
39. W. Mishels, D. Roose, Int. J. Bifurc. Chaos **12**, 1309 (2002)
40. H. Huijberts, W. Michiels, H. Nijmeijer, SIAM J. Appl. Dyn. Syst. **8**, 1 (2009)
41. C. Bick, M. Timme, C. Kolodziejski, SIAM J. Appl. Dyn. Syst. **4**, 1310 (2012)
42. E.N. Lorenz, J. Atmos. Sci. **20**, 130 (1963)
43. O.E. Rössler, Phys. Lett. A **57**, 397 (1976)
44. K. Engelborghs, T. Luzyanina, G. Samaey, Tech. Rep., Departament of Computer Science, K.U. Leuven, 2001
45. C.G. Broyden, Math. Comput. **19**, 577 (1965)
46. K. Pyragas, Phys. Rev. E **66**, 026207 (2002)
47. D.J. Tweten, B.P. Mann, Phys. Rev. E **86**, 046214 (2012)