Nonlinear Dyn (2012) 67:517–526
DOI 10.1007/s11071-011-0001-y

ORIGINAL PAPER

Computation of phase response curves via a direct method adapted to infinitesimal perturbations

Viktor Novičenko · Kestutis Pyragas

Received: 6 December 2010 / Accepted: 24 February 2011 / Published online: 16 March 2011

Abstract A new numerical algorithm for computation of phase response curves of stable limit cycle oscillators is proposed. The idea of the algorithm originates from a direct method that is based on computation of the oscillator response to short finite pulses delivered at different phases of oscillations. Here we adapt the direct method to the case of infinitesimal perturbations and compare our algorithm with the standard algorithm based on the backward integration of the adjoint equations. In contrast to the standard algorithm, our algorithm does not require any backward integration and it is easier to program since a necessity of numerical interpolation for the Jacobian matrix is avoided. In addition, we demonstrate by examples that our algorithm is faster than the standard algorithm and this advantage is especially notable for weakly stable limit cycle oscillators.

Keywords Phase response curve · Limit cycle oscillator · Neuronal model · Phase reduction method · Adjoint method

V. Novičenko (✉) · K. Pyragas
Center for Physical Sciences and Technology,
Semiconductor Physics Institute, 01108 Vilnius, Lithuania
e-mail: novicenko@pfi.lt

K. Pyragas
Department of Theoretical Physics, Faculty of Physics,
Vilnius University, 10222 Vilnius, Lithuania
e-mail: pyragas@pfi.lt

1 Introduction

Many physical, chemical and biological systems show periodic activity. Mathematically they can be modeled by limit cycle oscillators. It is well known that oscillators subjected to external signals or signals from other oscillators may exhibit a variety of behaviors [1–3]. The analysis of such systems can be essentially simplified if the perturbing signals are small. A fundamental theoretical technique for studying weakly perturbed oscillator dynamics is the phase reduction method (e.g. [1, 2]). The method is based on the assumption that the oscillator is strongly attracted to its limit cycle, so that amplitude variations can be neglected and only phase variations need to be considered. According to this method, equations of any weakly perturbed limit cycle oscillator can be reduced to a single scalar equation that describes the dynamics of the phase variable. An important characteristic of the limit cycle oscillator resulting from the phase reduction procedure is its phase-response (or phase-resetting) curve (PRC). The PRC describes the phase shift of the oscillation in response to a perturbing pulse at each phase of the oscillation. Historically, the formalism of the PRC was introduced by Hastings and Sweeney [4] in the context of resetting the circadian rhythms [5] and later on applied to other oscillatory systems. Knowing the PRC one can easily predict the behavior of oscillator when it is subjected to any small external stimuli or signals from other oscillators. For example, the PRC allows one to predict entrainment to periodic stimuli and phase locking with other coupled oscillators [3]. Furthermore, the knowledge of PRC usually suffices to understand collective dynamical properties like coherent oscillations, traveling waves, and pattern formation [1, 2].

The PRC is of particular importance for cardiology and neuroscience. In cardiac systems, the concept of PRC is useful for understanding the initiation of various arrhythmias [6–8]. Phase reduced models considerably simplify the complexity of detailed biological models in neuroscience [9–12]. The knowledge of the PRC reveals the bifurcation type underlying the onset of repetitive neuron firing [11] and facilitates the study of interactions leading to neural synchrony [9, 10, 12]. The PRC is also useful in control problems when constructing various neuron stimulation algorithms [13, 14].

The PRC can be measured experimentally [15–17] or evaluated theoretically [12, 18–21]. In the latter case, one needs to know the model equations. For some simple oscillator models, the phase reduction procedure can be performed analytically and corresponding PRC can be obtained in an analytical form [18, 19, 21]. However, in most cases the analytical methods fail and it is necessary to resort to a numerical approach [12, 20]. There are two basic ideas for numerical computation of PRC. One of them is based on direct computation of the oscillator response to short pulses delivered at different phases of the oscillator [22]. This algorithm is simple, but inaccurate, because it deals with the perturbations of a finite amplitude. Another idea is to linearize the system around the stable limit cycle and solve the linear adjoint equations [9, 11, 23, 24]. Such an algorithm is non-trivial since the adjoint equations are unstable and their solution requires a backward integration. Nevertheless, this algorithm is currently a standard approach for PRC computation. For example, it is implemented in a popular package XPPAUT [23]. Here we propose a simple and accurate algorithm based on the adaptation of the direct method for the case of infinitesimal perturbations. In addition to simplicity, our algorithm may have the advantage of being faster than the standard adjoint algorithm, especially when computing PRCs of weakly stable limit cycle oscillators.

The rest of the paper is organized as follows. In Sect. 2 we introduce the necessary definitions and discuss briefly the known algorithms of PRC computation. The description of our algorithm is presented in Sect. 3. In Sect. 4 we demonstrate our algorithm by several specific examples and compare it with the standard algorithm. The paper is finished by conclusions presented in Sect. 5.

2 PRC definition and algorithms of its computation

Let a dynamical system be defined by ordinary differential equations

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}), \tag{1}$$

where $\mathbf{X} = (x_1, x_2, \ldots, x_N)$ represents the vector of state variables in an N-dimensional phase space. Denote by $\mathbf{X}(t) = \mathbf{\Psi}(\mathbf{X}_0, t)$ the solution of system (1), where $\mathbf{X}_0 = \mathbf{X}(0)$ is the initial condition. Suppose that the system has an exponentially stable limit cycle with a period T. Choosing the initial condition on the limit cycle, $\mathbf{X}_0 = \mathbf{X}_c^0$, we get the T-periodic trajectory moving along the limit cycle: $\mathbf{X}_c(t) = \mathbf{\Psi}(\mathbf{X}_c^0, t)$, $\mathbf{\Psi}(\mathbf{X}_c^0, t+T) = \mathbf{\Psi}(\mathbf{X}_c^0, t)$. The phases of the points on the limit cycle can be defined in different ways: as a length of the limit cycle, as an angle variable in a polar coordinate system (for two-dimensional systems) or as a variable uniformly increasing in time when the phase point moves along the limit cycle. In the phase reduction method, the latter definition is a standard one since it leads to the most simple phase equation. Specifically, if we assign the zero phase for the point $\mathbf{X}_c^0$, then the phases of other points on the limit cycle are

$$\varphi\big(\mathbf{X}_c(t)\big) = t (\mathrm{mod}\ T). \tag{2}$$

The phase varies in the interval $[0, T]$ and satisfies a simple differential equation $\dot{\varphi}(\mathbf{X}_c(t)) = 1$. The phase of oscillation can be also introduced outside the limit cycle [28, 29]. Any trajectory $\mathbf{X}(t) = \mathbf{\Psi}(\mathbf{X}_0, t)$ with an initial condition $\mathbf{X}_0$ lying in the domain of attraction of the limit cycle approaches the limit cycle as $t \to \infty$. The phase $\varphi(\mathbf{X}_0)$ of the point $\mathbf{X}_0$ is defined as the phase of the asymptotic solution, i.e., $\mathbf{X}(t) \to \mathbf{X}_c(t + \varphi(\mathbf{X}_0))$ for $t \to \infty$. Thus any point in the domain of attraction of the limit cycle has a phase. The set of all points $\mathbf{X}$ with equal phases $\varphi(\mathbf{X}) = \mathit{constant}$ forms an $(N-1)$-dimensional manifold referred to as an isochron. Such a global definition of the phases and isochrons allows one to predict the phase shift of the oscillator caused by any strong pulsed perturbation. An extension of the PRC definition to a neighborhood of the limit cycle can be found in the recent publication [21]. Based on parameterization method for invariant manifolds and Lie symmetries, the authors introduce a so-called "phase response surface." However, here we restrict ourselves to consideration of weak perturbations.

Consider a weakly perturbed system (1):

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}) + \varepsilon \mathbf{G}(t), \tag{3}$$

where $\mathbf{G}(t)$ is a vector of time-depended perturbation and ε is a small parameter. Following the Kuramoto's [1] approach the system can be reduced to a phase equation as follows. Consider first the unperturbed ($\varepsilon = 0$) oscillator (3), and differentiate over the time the function $\varphi(\mathbf{X})$ that describes the phases of points near the limit cycle. Using the chain rule we obtain $\dot{\varphi}(\mathbf{X}) = \nabla\varphi(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X})$, where the dot between the vectors denotes their scalar product. On the other hand, the phase satisfies the equation $\dot{\varphi}(\mathbf{X}) = 1$, exactly the same as on the limit cycle, since isochrons are mapped to isochrons by the flow of the vector field $\mathbf{F}(\mathbf{X})$. Thus it follows a useful equality:

$$\nabla\varphi(\mathbf{X}) \cdot \mathbf{F}(\mathbf{X}) = 1. \tag{4}$$

Applying the chain rule to (3) for $\varepsilon \neq 0$ and using (4), one obtains the phase equation for the perturbed system:

$$\dot{\varphi} = 1 + \varepsilon \mathbf{G}(t) \cdot \nabla\varphi\bigl(\mathbf{X}(t)\bigr). \tag{5}$$

For small perturbations, the trajectory of the system remains close to the limit cycle and thus the gradient of the phase in (5) can be evaluated on the point $\mathbf{X}_c(\varphi)$ of the limit cycle lying on the same isochron as the point $\mathbf{X}(t)$, $\nabla\varphi(\mathbf{X}(t)) \approx \nabla\varphi(\mathbf{X}_c(\varphi))$. This T-periodic vector function is called the infinitesimal PRC, or simply the PRC. Its jth component is defined as

$$\mathbf{PRC}_j(\varphi) = \frac{\partial\varphi(\mathbf{X}_c(\varphi))}{\partial x_j}. \tag{6}$$

Finally, the phase reduced model of the perturbed system (3) to first order in ε is given by

$$\dot{\varphi} = 1 + \varepsilon \mathbf{G}(t) \cdot \mathbf{PRC}(\varphi). \tag{7}$$

The PRC contains all necessary information about the dynamical system, which allows us to predict the behavior of the phase in the presence of any small time-depended perturbation. In the following sections we discuss briefly two main numerical methods currently used for PRC computation.

2.1 Direct method

This method [22] is based on direct computation of the system response to a simple class of specific perturbations. Suppose that the initial condition of the unperturbed system (1) is placed on the limit cycle with some defined phase φ: $\mathbf{X}_0 = \mathbf{X}_c(\varphi)$. Now add a small perturbation Δx_j to the jth component of the initial vector $\mathbf{X}_0$, leaving other components unperturbed. According to the PRC definition (6), the phase of the perturbed systems changes by the amount $\Delta x_j \mathbf{PRC}_j(\varphi)$ provided Δx_j is small. This means that the solution of the system (1) with the perturbed initial condition converges to $\mathbf{X}(t) \to \mathbf{X}_c(\varphi + \Delta x_j \mathbf{PRC}_j(\varphi) + t)$ as $t \to \infty$.

In practice, this algorithm is realized as follows. For a given φ, the system (1) is integrated for two different initial conditions, one taken on the limit cycle $\mathbf{X}_c(\varphi)$ and another shifted by a distance Δx_j in the direction of the jth component of the vector field. After sufficiently long time, the second solution approaches the limit cycle and then one estimates the phase difference $\Delta\varphi(\varphi)$ between these two solutions. The PRC is evaluated as $\mathbf{PRC}_j(\varphi) = \Delta\varphi(\varphi)/\Delta x_j$. This algorithm provides probably the most simple way for numerical estimation of the PRC. However, this algorithm is not accurate since it operates with the finite perturbation Δx_j, whereas the PRC is defined for the infinitesimal perturbations.

2.2 Adjoint method

The adjoint method is based on the Malkin's approach [25–27], which gives an alternative way of derivation of the phase reduced equation. According to this approach, the phase dynamics of the perturbed system (3) in the first order to ε is given by

$$\dot{\varphi} = 1 + \varepsilon \mathbf{G}(t) \cdot \mathbf{Q}(\varphi), \tag{8}$$

where the T-periodic vector function $\mathbf{Q}$ satisfies the linear adjoint equation

$$\dot{\mathbf{Q}} = -\bigl\{D\mathbf{F}\bigl(\mathbf{X}_c(t)\bigr)\bigr\}^T \mathbf{Q} \tag{9}$$

with the initial condition $\mathbf{Q}(0)\cdot\mathbf{F}(\mathbf{X}_c(0))=1$. Here $\{D\mathbf{F}(\mathbf{X}_c(t))\}^T$ is the transposed Jacobian of $\mathbf{F}$ at the point $\mathbf{X}_c(t)$ on the limit cycle. Note that the condition $\mathbf{Q}(t)\cdot\mathbf{F}(\mathbf{X}_c(t))=1$ holds for any t, because it is easy to show that $\frac{d}{dt}\{\mathbf{Q}(t)\cdot\mathbf{F}(\mathbf{X}_c(t))\}=0$.

Comparing (7) and (8) we see that $\mathbf{PRC}(\varphi)=\mathbf{Q}(\varphi)$. Thus (9) provides an alternative way for numerical computation of the PRC. Unfortunately, the solution of the adjoint equation (9) is not a trivial problem. Due to the periodicity requirement $\mathbf{Q}(T)=\mathbf{Q}(0)$, we encounter a boundary value problem. Since the Jacobian in (9) depends on the periodic orbit $\mathbf{X}_c(t)$, it seems rational to integrate (9) together with (1) by choosing its initial condition on the periodic orbit. However, such a direct integration is impossible since the adjoint equation (9) is unstable. Ermentrout [9, 11] proposed to integrate (9) backwards in time and implemented this idea in the XPPAUT package [23]. When integrating backwards, the solution of (9) approaches the periodic solution corresponding to the PRC. However, the backward integration of (9) together with (1) is impossible since now the periodic solution $\mathbf{X}_c(t)$ becomes unstable. Therefore, when integrating (9) backwards in time some numerical interpolation of the Jacobian matrix is needed. The latter has to be obtained from the forward integration of (1). Another shortcoming of this algorithm is that the solution of the adjoint equation (9) converges slowly to the PRC when the limit cycle is weakly stable. The adjoint method is currently the standard algorithm for PRC computation, therefore we compare it with our algorithm in Sect. 5.

Note that Govaerts and Sautois [24] have recently proposed a modification for the adjoint method. Here the PRC is obtained as a by-product of the boundary value problem for the limit cycle of the stable orbit. The boundary value problem is solved by collocation at Gauss points. The method starts from an initial guess of the periodic orbit and then uses an iteration procedure in order to obtain the true solution. The method is useful in the context of the numerical continuation of orbits with a variable parameter of the system. The authors stress that “the method is not very relevant if only one or a few PRCs are needed.” This differs from our proposed method, which is efficient for computation of a single PRC for a fixed value of a parameter.

3 Adaptation of the direct method to infinitesimal perturbations

To adapt the direct method to the infinitesimal perturbations we employ the variational equation

$$\delta\dot{\mathbf{X}}=D\mathbf{F}\big(\mathbf{X}_c(t+\varphi)\big)\delta\mathbf{X} \tag{10}$$

that describes the dynamics of infinitesimal deviations from the limit cycle. Here $\mathbf{X}_c(t+\varphi)$ is the limit cycle solution of the system (1) with the initial condition $\mathbf{X}_0=\mathbf{X}_c(\varphi)$ that represents the point on the limit cycle with the phase φ. Note that (10) can be integrated together with (1) since they both are stable. Thus contrary to the adjoint method here we do not need any numerical interpolation of the Jacobian in (10).

To obtain the jth component of the PRC at the phase φ we choose the initial condition for the variational equation (10) as $\delta x_k(0)=\delta_{jk}$, where δ_{jk} is the Kronecker's delta. This means that the initial vector $\delta\mathbf{X}(0)$ has all zero components except j that is equal to one. Let us now integrate (10) for an integer number p of periods and obtain $\delta\mathbf{X}(pT)$. Due to the stability of the limit cycle the vector $\delta\mathbf{X}(pT)$ becomes parallel to the velocity vector $\mathbf{V}(\varphi)\equiv\dot{\mathbf{X}}(\varphi)=\mathbf{F}(\mathbf{X}_c(\varphi))$ as $p\to\infty$. The $\mathbf{PRC}_j(\varphi)$ by definition is equal to the phase shift of the perturbed trajectory at the point $\mathbf{X}_c(\varphi)$, i.e., $\lim_{p\to\infty}\delta\mathbf{X}(pT)=\mathbf{V}(\varphi)\mathbf{PRC}_j(\varphi)$. Alternatively, this equality can be written as follows:

$$\mathbf{PRC}_j(\varphi)=\lim_{p\to\infty}\frac{\mathbf{V}^T(\varphi)\delta\mathbf{X}(pT)}{\mathbf{V}^T(\varphi)\mathbf{V}(\varphi)}. \tag{11}$$

This expression allows us to compute the PRC by a simple forward integration of the variational equation (10) together with (1) that defines the limit cycle solution.

This algorithm can be yet improved. According to (11) we have to integrate the above system of equations for several periods p, until the deviation vector $\delta\mathbf{X}(pT)$ becomes parallel to the velocity vector $\mathbf{V}(\varphi)$. The number of required periods can be extremely large if the limit cycle is weakly stable. We can avoid the necessity of integration for several periods if instead of variational equation (10) we employ the equation for the fundamental matrix. It suffices to know the fundamental matrix solution for only one period of the limit cycle in order to extract the information on the PRC. Below we describe the algorithm for the computation of the fundamental matrix and subsequent extraction of the PRC from this matrix.

3.1 Computation of the fundamental matrix

To compute the PRC at different phases φ of the limit cycle, we define the $N \times N$ fundamental matrix $\mathbf{\Phi}_\varphi(t)$ governed by the differential equation

$$\dot{\mathbf{\Phi}}_\varphi(t) = D\mathbf{F}\big(\mathbf{X}_c(t+\varphi)\big)\mathbf{\Phi}_\varphi(t) \tag{12}$$

with the initial condition $\mathbf{\Phi}_\varphi(0) = \mathbf{I}_N$, where $\mathbf{I}_N$ is the $N \times N$ identity matrix. The PRC computation requires the knowledge of the fundamental matrix for a given φ at $t = T$, i.e., $\mathbf{\Phi}_\varphi(T)$. Let us split the period T into n equal intervals $\Delta t = T/n \equiv \Delta\varphi$ and define n nodal points $\mathbf{X}_c(\varphi_i)$ on the limit cycle with the equally spaced phases $\varphi_i = i\Delta\varphi$, $i = 0, \ldots, n-1$. Let us also denote $\mathbf{\Phi}_{\varphi_i}(t) \equiv \mathbf{\Phi}_i(t)$. To determine $\mathbf{\Phi}_i(T)$ for any nodal point, it suffices to integrate (12) in small time intervals $t \in [0, \Delta\varphi]$ between the neighboring nodal points and obtain a sequence of the auxiliary matrixes $\mathbf{\Phi}_0(\Delta\varphi)$, $\mathbf{\Phi}_1(\Delta\varphi)$, $\ldots$, $\mathbf{\Phi}_{n-1}(\Delta\varphi)$. Then the desired matrix $\mathbf{\Phi}_i(T)$ at any nodal point i can be computed as a product of the above matrixes:

$$\begin{aligned}\mathbf{\Phi}_i(T) = {} & \mathbf{\Phi}_{i-1}(\Delta\varphi)\cdots\mathbf{\Phi}_0(\Delta\varphi)\\ & \times \mathbf{\Phi}_{n-1}(\Delta\varphi)\cdots\mathbf{\Phi}_i(\Delta\varphi).\end{aligned} \tag{13}$$

Note that the total time of integration of (12) when computing all auxiliary matrixes $\mathbf{\Phi}_i(\Delta\varphi)$ is equal to the period of the limit cycle. The required matrixes $\mathbf{\Phi}_i(T)$ are obtained by multiplication of the matrixes $\mathbf{\Phi}_i(\Delta\varphi)$ in different sequences according to (13). In Sect. 3.2 we show how to extract the value $\mathbf{PRC}(\varphi_i)$ from the determined matrix $\mathbf{\Phi}_i(T)$.

3.2 Extracting the PRC from the fundamental matrix

We now describe the procedure of determination of the PRC at the ith nodal point from the known matrix $\mathbf{\Phi}_i(T)$. To simplify the description we omit the index i in the expressions presented below, but we keep in mind that it is valid for any phase φ.

Using the fundamental matrix $\mathbf{\Phi}(T)$, the deviation $\delta\mathbf{X}(pT)$ in (11) can be presented in the form

$$\delta\mathbf{X}(pT) = \mathbf{\Phi}^p(T)\delta\mathbf{X}(0). \tag{14}$$

We should remind that all components of the initial perturbation $\delta\mathbf{X}(0)$ are equal to zero, except the jth component, which is equal to one. It means that here we deal with the jth component of the PRC. To simplify expression (14) we use the method of spectral decomposition based on the Floquet theory. We suppose that the fundamental matrix is nonsingular and define its right $\mathbf{R}_k$ and left $\mathbf{L}_k$ eigenvectors:

$$\mathbf{\Phi}(T)\mathbf{R}_k = \mu_k\mathbf{R}_k, \tag{15}$$

$$\mathbf{L}_k^T\mathbf{\Phi}(T) = \mu_k\mathbf{L}_k^T, \tag{16}$$

where μ_k, $k = 1, \ldots, N$, are the Floquet multipliers of the limit cycle. They satisfy the characteristic equation

$$\det\big[\mathbf{\Phi}(T) - \mu\mathbf{I}_N\big] = 0. \tag{17}$$

One of the multiplies that describes an evolution of small deviations along the limit cycle is equal to one, $\mu_1 = 1$. The absolute values of other multipliers of the stable limit are less than one:

$$|\mu_N| < |\mu_{N-1}| < \cdots < |\mu_2| < 1. \tag{18}$$

The left and right eigenvectors corresponding to different multipliers are orthogonal to each other:

$$\mathbf{L}_k^T\mathbf{R}_l = 0, \quad \text{when } l \neq k. \tag{19}$$

We now expand the initial perturbation $\delta\mathbf{X}(0)$ in terms of the right eigenvectors,

$$\delta\mathbf{X}(0) = c_1\mathbf{R}_1 + \cdots + c_N\mathbf{R}_N, \tag{20}$$

and substitute this expression into (14). Due to the inequalities (18), (14) in the limit of large p transforms to

$$\lim_{p\to\infty}\delta\mathbf{X}(pT) = \lim_{p\to\infty}\sum_{k=1}^{N}\mu_k^p c_k\mathbf{R}_k = c_1\mathbf{R}_1. \tag{21}$$

The right eigenvector corresponding to the first multiplier $\mu_1 = 1$ can be chosen equal to the velocity vector, $\mathbf{R}_1 = \mathbf{V}$, since $\mathbf{V}$ satisfies an obvious equality $\mathbf{\Phi}(T)\mathbf{V} = \mathbf{V}$. Then substituting (21) into (11) we obtain that the PRC is equal to the first coefficient in the expansion (20):

$$\mathbf{PRC}_j = c_1. \tag{22}$$

We now multiply (20) by the left eigenvector $\mathbf{L}_1^T$ and due to the orthogonality property (19) we obtain

$\mathbf{L}_1^T\delta\mathbf{X}(0) = c_1\mathbf{L}_1^T\mathbf{R}_1$. Substituting $\mathbf{R}_1 = \mathbf{V}$ and using (22) we get

$$\mathbf{PRC}_j = \frac{\mathbf{L}_1^T\delta\mathbf{X}(0)}{\mathbf{L}_1^T\mathbf{V}}. \tag{23}$$

Since the initial perturbation $\delta\mathbf{X}(0)$ has only jth nonzero component, the numerator in this equation can be simplified to $\mathbf{L}_1^T\delta\mathbf{X}(0) = \mathbf{L}_1[j]$, where $\mathbf{L}_1[j]$ is the jth component of the vector $\mathbf{L}_1$. Thus the jth component of the vector **PRC** is related to the jth component of the vector $\mathbf{L}_1$. It means that the entire **PRC** vector is proportional to the left eigenvector $\mathbf{L}_1$ and we obtain the final equation for the PRC vector in the form

$$\mathbf{PRC}(\varphi) = \frac{\mathbf{L}_1(\varphi)}{\mathbf{L}_1^T(\varphi)\mathbf{V}(\varphi)}. \tag{24}$$

This equation constitutes the basis of our algorithm. We see that the problem of PRC computation reduces to the problem of evaluation of the left eigenvector $\mathbf{L}_1$ that satisfies the matrix equation $\mathbf{L}_1^T[\mathbf{\Phi}(T) - \mathbf{I}_N] = 0$. Because the determinant of this system is equal to zero, the value of one of components of the vector $\mathbf{L}_1^T$ can be assigned arbitrarily. We choose the first component equal to one, $\mathbf{L}_1^T[1] = 1$. Then the other components $\mathbf{L}_1^T[2:N]$ are obtained by solving the reduced system of $N-1$ linear equations:

$$\mathbf{L}_1^T[2:N]\{\mathbf{\Phi}(T)[2:N;2:N] - \mathbf{I}_{N-1}\} = -\mathbf{\Phi}(T)[1;2:N]. \tag{25}$$

Here $\mathbf{\Phi}(T)[2:N;2:N]$ is a submatrix of the matrix $\mathbf{\Phi}(T)$ formed by removing the first row and the first column of the original matrix.

We use (24) to evaluate the PRC for the phases at the nodal points $\varphi_i = i\Delta\varphi$, where the values of the fundamental matrix $\mathbf{\Phi}_i(T)$ are defined. The nodal points do not need to be taken very densely. To compute the values of the PRC for the phases between the nodal points, we can use the adjoint equation (9). Specifically, the PRC for the phases in the interval $\varphi \in [\varphi_i, \varphi_{i+1}]$ can be determined by integrating the adjoint equation

$$\frac{d}{d\varphi}\mathbf{Q}(\varphi) = -\{D\mathbf{F}(\mathbf{X}_c(\varphi))\}^T\mathbf{Q}(\varphi) \tag{26}$$

with the initial condition $\mathbf{Q}(\varphi_i) = \mathbf{PRC}(\varphi_i)$ at the nodal point φ_i determined from (24). Although (26) is unstable, in a small interval $[\varphi_i, \varphi_i + \Delta\varphi]$ it can be integrated forwards in time. For sufficiently small $\Delta\varphi$, the value $\mathbf{Q}(\varphi_{i+1})$ on the end φ_{i+1} of the interval obtained from the adjoint equation (26) almost coincides with the value $\mathbf{PRC}(\varphi_{i+1})$ in the next nodal point determined from (24), $\mathbf{Q}(\varphi_{i+1}) \approx \mathbf{PRC}(\varphi_{i+1})$. Note that this step of the algorithm is not mandatory. The integration between the nodal points can be avoided if they are chosen sufficiently dense. However, the increase of the number n of the nodal points reduces the speed of the algorithm.

3.3 Main steps of the algorithm

To summarize our algorithm we list the main steps.

Step 1. Define n nodal points on the limit cycle with equally spaced phases $\varphi_i = i\Delta\varphi$ and compute the auxiliary matrixes $\mathbf{\Phi}_i(\Delta\varphi)$ by integrating (12) and (1) in the interval $t \in [0, \Delta\varphi]$ for different $\varphi = \varphi_i$.

Step 2. At each nodal point, evaluate the fundamental matrix $\mathbf{\Phi}_i(T)$. To this end, multiply the auxiliary matrixes $\mathbf{\Phi}_i(\Delta\varphi)$ in different sequences, as prescribed by (13).

Step 3. Solve the linear system (25) to determine the left eigenvector $\mathbf{L}_1(\varphi_i)$ and using (24) evaluate $\mathbf{PRC}(\varphi_i)$ at each nodal point.

Step 4. If one needs to know the values of the PRC between the nodal points, then integrate forwards the adjoint equation (26) with the predefined (in Step 3) initial conditions at the nodal points.

4 Numerical demonstrations

We have tested our algorithm for a variety of dynamical systems and compared its speed with that of the standard algorithm based on the backward integration of the adjoint equation. Here we present the results for three dynamical systems: the Morris–Lecar [30], the Hodgin–Huxley [31] and the Hindmarsh–Rose [32] neuron models. The equations of these models as well as the values of the parameters are presented in the Appendix. Both algorithms where realized in the MATLAB environment and the computations were performed on the same computer (2.8 GHz CPU and 4.0 GB RAM). All differential equations were solved by the Runge–Kutta 4,5 method using the `ode45` solver with the precision 10^{-9}.

The number n of the nodal points in our algorithm was chosen such that to guarantee a given precision of the solution of the adjoint equation (26) in the intervals between the nodal points. Specifically, we estimated the differences $\epsilon_i = |\mathbf{Q}(\varphi_i) - \mathbf{PRC}(\varphi_i)|$, where $\mathbf{Q}(\varphi_i)$ is the solution of (26) on the end of the interval $[\varphi_{i-1}, \varphi_i]$ with the initial condition $\mathbf{Q}(\varphi_{i-1}) = \mathbf{PRC}(\varphi_{i-1})$, with $\mathbf{PRC}(\varphi_i)$ being the PRC value on the ith nodal point determined from (24). We chose the minimal number of nodal points in such a way as to provide the precision $\epsilon_i < 10^{-2}$ for all nodal points.

In the standard algorithm based on the backward integration of the adjoint equation (9), we used the cubic spline data interpolation of the Jacobian matrix. To define a precision of the adjoint method, we estimated the difference $|\mathbf{Q}(pT) - \mathbf{Q}((p-1)T)|$, where p is the number of periods of the backward integration. The integration was stopped when this difference decreased to the value less than 10^{-2}.

When estimating the computation time of both algorithms, the time needed for the calculation of the limit cycle has been excluded. We have assumed that the initial conditions and the period of the limit cycle are known at the start point and have estimated only the time needed for the PRC computation.

For neuron models, only the first component of the PRC vector is of interest since perturbations are usually applied only to the first equation that describes the dynamics of the membrane potential. Therefore, in all figures presented below we show only the curve $\mathbf{PRC}_1(\varphi)$. Both algorithms lead to the same results which in the figures are indistinguishable. However, our algorithm is 10–100 times faster than the standard algorithm.

In Fig. 1 we show the membrane potential and the PRC for the Morris–Lecar neuron model. The computation time needed to reproduce these results with our algorithm is 10.3 times less than that needed for the standard algorithm. Similar results for the Hodgkin–Huxley neuron model are presented in Fig. 2. Here the parameters are chosen in such a way that the system possesses two stable limit cycles. The (V, m) projections of both limit cycles are shown in Fig. 2(a), while the PRCs for these limit cycles are presented in Fig. 2, (b) and (c). Again, for this system our algorithm is 36.9 times faster than the standard algorithm.

The advantages of our algorithm are particularly notable for the case when the limit cycle is weakly stable. This may happen when the dynamical system is close to a bifurcation point. Here we demonstrate such a situation for the Hodgkin–Huxley model when the parameters are close the saddle-node bifurcation of the limit cycles. In Fig. 3 we show the dependence of the computation time on the bifurcation parameter I_0 (an external dc current) when computing the PRC with the standard algorithm. Close to the bifurcation point $I_0^* \approx 9.84\ \mu\text{A/cm}^2$ the computation time increases drastically to the values of 10^3 s. The reason for inefficiency of the adjoint method is that the convergence rate to the periodic solution of the adjoint equation during the backward integration is defined by the leading Floquet multiplier of the limit cycle, such that the convergence rate is directly related to the stability of the limit cycle. In contrast, our algorithm is independent of the bifurcation parameter; the computation time for our algorithm is equal approximately to 2 s for any value of the bifurcation parameter I_0.

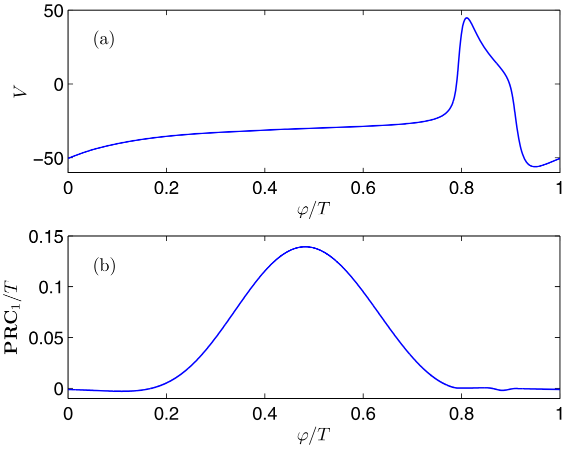

Fig. 1 The membrane potential (**a**) and the first component of the PRC (**b**) for the Morris–Lecar neuron model. Our algorithm is 10.3 times faster than the standard algorithm. The number of the nodal points is $n = 100$

Finally, we tested our algorithm for a complex limit cycle that appears in the Hindmarsh–Rose neuron model in the bursting regime. The results are presented in Fig. 4. For this model, our algorithm is 74.2 times faster than the standard algorithm.

5 Conclusions

In this paper, we have proposed a new algorithm for numerical computation of the phase response curves of the stable limit cycles. The idea of the algorithm is

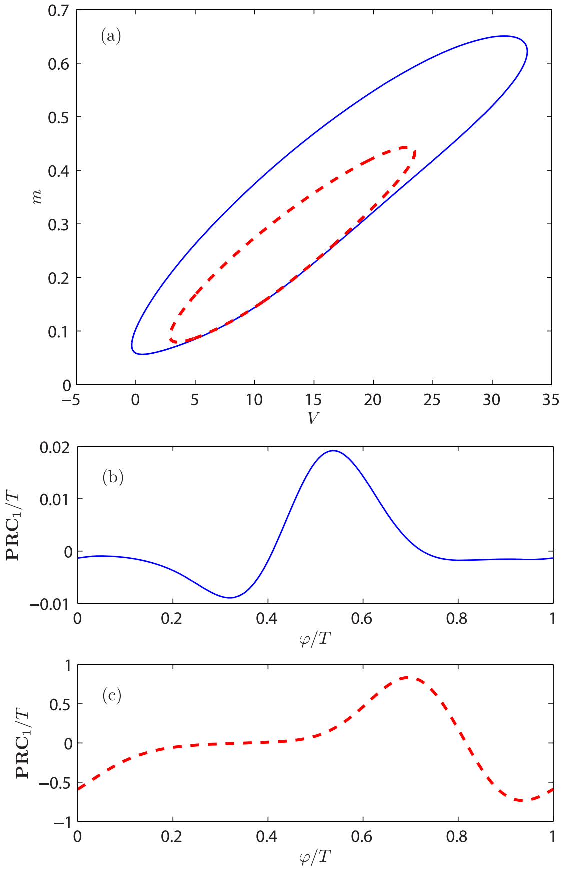

Fig. 2 The (V, m) projections of two coexisting stable limit cycles of the Hodgkin–Huxley neuron model (**a**) and the corresponding PRCs (**b**) and (**c**). Our algorithm is 36.9 times faster than the standard algorithm. The number of the nodal points is $n = 100$

based on the adaptation of the direct method to the infinitesimal perturbations. Contrary to the standard algorithm based on the backward integration of the linear adjoint equations, our algorithm does not require the backward integration and as a result we avoid the necessity of numerical interpolation of the Jacobian matrix. By testing our algorithm for several neuron models, we have shown that it is 10–100 times faster than the standard algorithm. The algorithm has particular advantage when computing the phase response curves of weakly stable oscillators, e.g. close to bifurcation points of the limit cycles. In such a case, the standard algorithm may become very slow, whereas the speed of our algorithm is independent of the bifurcation parameter. Another advantage of our algorithm is that it can be readily adapted for parallel computation. All steps of our algorithm can be easily parallelized since the computations at the different nodal points are independent of each other.

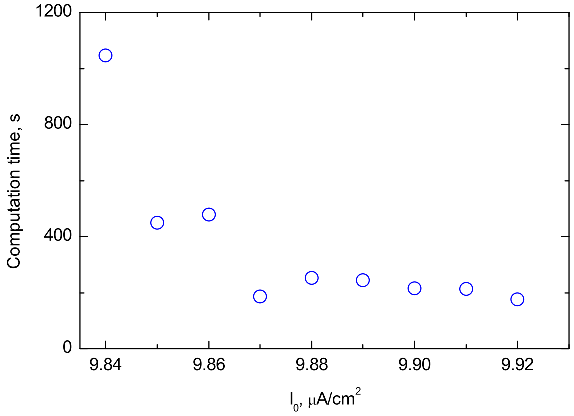

Fig. 3 The computation time of the PRC via the standard algorithm versus the bifurcation parameter I_0 for the Hodgin–Huxley neuron model. The computation time of our algorithm is independent of the bifurcation parameter and is equal approximately to 2 s. For all PRCs, the number of the nodal points is $n = 100$

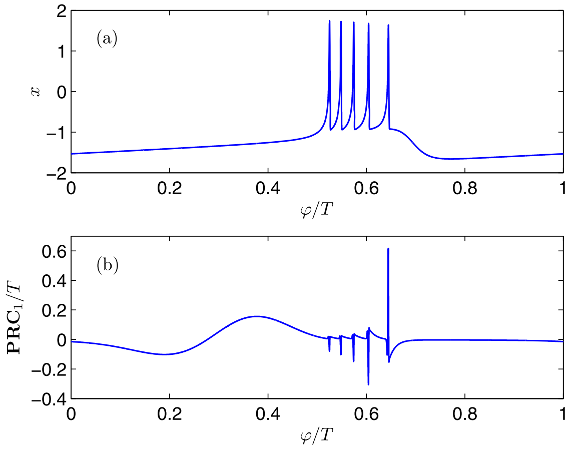

Fig. 4 The membrane potential (**a**) and the first component of the PRC (**b**) for the Hindmarsh–Rose neuron model. Our algorithm is 74.2 times faster than the standard algorithm. The number of the nodal points is $n = 1000$

Acknowledgements We acknowledge the support from the Global Grant No. VP1-3.1-ŠMM-07-K-01-025.

Appendix

6.1 Morris–Lecar neuron model

The equations of the Morris–Lecar model are [30]:

$$C\dot{V} = -g_{Ca} m_\infty(V)(V - V_{Ca}) - g_K \omega (V - V_K) - g_l(V - V_l) + I_0, \quad (27)$$
$$\dot{\omega} = \phi\big[\omega_\infty(V) - \omega\big]/\tau_\omega(V),$$

where

$$m_\infty(V) = 0.5\big\{1 + \tanh\big[(V - V_1)/V_2\big]\big\},$$
$$\omega_\infty(V) = 0.5\big\{1 + \tanh\big[(V - V_3)/V_4\big]\big\}, \quad (28)$$
$$\tau_\omega(V) = 1/\cosh\big[(V - V_3)/(2V_4)\big].$$

The values of the parameters are: $C = 5.0\ \mu\text{F/cm}^2$, $g_{Ca} = 4.0\ \mu\text{S/cm}^2$, $g_K = 8.0\ \mu\text{S/cm}^2$, $g_l = 2.0\ \mu\text{S/cm}^2$, $V_{Ca} = 120$ mV, $V_K = -80$ mV, $V_l = -60$ mV, $V_1 = -1.2$ mV, $V_2 = 18.0$ mV, $V_3 = 12.0$ mV, $V_4 = 17.4$ mV, $\phi = 1/15\ \text{s}^{-1}$, $I_0 = 40\ \mu\text{A/cm}^2$.

6.2 Hodgkin–Huxley neuron model

The equations of the Hodgkin–Huxley model are [31]:

$$C\dot{V} = -g_{Na} m^3 h(V - V_{Na}) - g_K n^4 (V - V_K) - g_l(V - V_l) + I_0,$$
$$\dot{m} = \alpha_m(V)(1 - m) - \beta_m(V) m, \quad (29)$$
$$\dot{h} = \alpha_h(V)(1 - h) - \beta_h(V) h,$$
$$\dot{n} = \alpha_n(V)(1 - n) - \beta_n(V) n,$$

where

$$\alpha_m(V) = (2.5 - 0.1V)/\big[\exp(2.5 - 0.1V) - 1\big],$$
$$\beta_m(V) = 4\exp(-V/18),$$
$$\alpha_h(V) = 0.07\exp(-V/20),$$
$$\beta_h(V) = 1/\big[\exp(3 - 0.1V) + 1\big], \quad (30)$$
$$\alpha_n(V) = (0.1 - 0.01V)/\big[\exp(1 - 0.1V) - 1\big],$$
$$\beta_n(V) = 0.125\exp(-V/80).$$

The values of the parameters for the regime with two coexisting stable limit cycles (Fig. 2) are: $C = 1\ \mu\text{F/cm}^2$, $g_{Na} = 120\ \text{mS/cm}^2$, $g_K = 36\ \text{mS/cm}^2$, $g_l = 0.3\ \text{mS/cm}^2$, $V_{Na} = 85.7$ mV, $V_K = -11$ mV, $V_l = 10.559$ mV, $I_0 = 41\ \mu\text{A/cm}^2$. The regime close to a saddle-node bifurcation of the limit cycles (Fig. 3) is obtained with the same set of the parameters, except $V_{Na} = 115$ mV, $V_K = -40$ mV, and I_0 is taken as the bifurcation parameter.

6.3 Hindmarsh–Rose neuron model

The equations of the Hindmarsh–Rose model are [32]:

$$\dot{x} = y - ax^2 - x^3 - z + I,$$
$$\dot{y} = 1 - bx^2 - y, \quad (31)$$
$$\dot{z} = r\big[s(x - x_R) - z\big].$$

The values of the parameters corresponding to the bursting regime are: $a = 3$, $I = 1.3$, $b = 5$, $r = 0.001$, $s = 4$, $x_R = -1.6$.

References

1. Kuramoto, Y.: Chemical Oscillations, Waves, and Turbulence. Dover, New York (2003)
2. Pikovsky, A., Rosemblum, M., Kurths, J.: Synchronization: A Universal Concept in Nonlinear Science. Cambridge University Press, Cambridge (2001)
3. Winfree, A.: The Geometry of Biological Time. Springer, New York (2001)
4. Hastings, J.W., Sweeney, B.M.: A persistent diurnal rhythm of luminescence in Gonyaulax polyedra. Biol. Bull. **115**, 440–458 (1958)
5. Johnson, C.H.: Forty years of PRC—what have we learned? Chronobiol. Int. **16**, 711–743 (1999)
6. Winfree, A.T.: When Time Breaks Down. Princeton University Press, Princeton (1987)
7. Ikeda, N.: Model of bidirectional interaction between myocardial pacemakers based on the phase response curve. Biol. Cybern. **43**, 157–167 (1982)
8. Tsalikakisa, D.G., Zhangb, H.G., Fotiadisa, D.I., Kremmydasa, G.P., Michalis, L.K.: Phase response characteristics of sinoatrial node cells. Comput. Biol. Med. **37**, 8–20 (2007)
9. Ermentrout, G.B., Kopell, N.: Multiple pulse interactions and averaging in systems of coupled neural oscillators. J. Math. Biol. **29**, 195–217 (1991)
10. Ermentrout, G.B.: Stable periodic solutions to discrete and continuum arrays of weakly coupled nonlinear oscillators. SIAM J. Appl. Math. **52**, 1665–1687 (1992)
11. Ermentrout, B.: Type I membranes, phase resetting curves, and synchrony. Neural Comput. **8**, 979–1001 (1996)
12. Izhikevich, E.M.: Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting. MIT Press, Cambridge (2007)
13. Tass, P.: Phase Resetting in Medicine and Biology. Springer, Berlin (1999)
14. Stiger, T., Danzl, P., Moehlis, J., Netoff, T.I.: Linear control of neuronal spike timing using phase response curves. J. Med. Devices **4**, 027533 (2010)
15. Reyes, A.D., Fetz, E.E.: Two modes of interspike interval shortening by brief transient depolarizations in cat neocortical neurons. J. Neurophysiol. **69**, 1661–1672 (1993)
16. Galan, R.F., Ermentrout, G.B., Urban, N.N.: Efficient estimation of phase-resetting curves in real neurons and its significance for neural-network modeling. Phys. Rev. Lett. **94**, 158101 (2005)

17. Tateno, T., Robinson, H.P.C.: Phase resetting curves and oscillatory stability in interneurons of rat somatosensory cortex. Biophys. J. **92**, 683–695 (2007)
18. Brown, E., Moehlis, J., Holmes, P.: On the phase reduction and response dynamics of neural oscillator populations. Neural Comput. **16**, 673–715 (2004)
19. Kuznetsov, A.P., Stankevich, N.V., Turukina, L.V.: Coupled van der Pol–Duffing oscillators: phase dynamics and structure of synchronization tongues. Physica D **238**, 1203–1215 (2009)
20. Sherwood, E.W., Guckenheimer, J.: Dissecting the phase response of a model bursting neuron. SIAM J. Appl. Dyn. Syst. **9**, 659–703 (2010)
21. Guillamon, A., Huguet, G.: A computational and geometric approach to phase resetting curves and surfaces. SIAM J. Appl. Dyn. Syst. **8**, 1005–1042 (2009)
22. Guevara, M.R., Glass, L., Mackey, M.C., Shier, A.: Chaos in neurobiology. IEEE Trans. Syst. Man Cybern. **13**(5), 790–798 (1983)
23. Ermentrout, G.: Simulating, Analysing, and Animating Dynamical Systems: A Guide to XXPAUT for Researchers and Students. SIAM, Philadelphia (2002)
24. Govaerts, W., Sautois, B.: Computation of the phase response curve: a direct numerical approach. Neural Comput. **18**, 817–847 (2006)
25. Malkin, I.G.: Methods of Poincaré and Lyapunov in Theory of Non-linear Oscillations. Gostexizdat, Moscow (1949) (in Russian: Metodi Puankare i Liapunova v teorii nelineinix kolebanii)
26. Malkin, I.G.: Some Problems in Nonlinear Oscillation Theory. Gostexizdat, Moscow (1956) (in Russian: Nekotorye zadachi teorii nelineinix kolebanii)
27. Hoppensteadt, F.C., Izhikevich, E.M.: Weakly Connected Neural Networks. Springer, New York (1997)
28. Winfree, A.: Patterns of phase compromise in biological cycles. J. Math. Biol. **1**, 73–95 (1974)
29. Guckenheimer, J.: Isochrons and phaseless sets. J. Math. Biol. **1**, 259–273 (1975)
30. Morris, C., Lecar, H.: Voltage oscillations in the barnacle giant muscle fiber. Biophys. J. **35**, 193–213 (1981)
31. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. **117**, 500–544 (1952)
32. Hindmarsh, J.L., Rose, R.M.: A model of neuronal bursting using three coupled first order differential equations. Proc. R. Soc. Lond. B, Biol. Sci. **221**, 87–102 (1984)