Computation, continuation and bifurcation analysis of periodic solutions of delay differential equations

Tatyana Luzyanina Koen Engelborghs Kurt Lust
Dirk Roose

Report TW 252, January 1997

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Computation, continuation and bifurcation analysis of periodic solutions of delay differential equations*

Tatyana Luzyanina† *Koen Engelborghs*‡ *Kurt Lust*‡
Dirk Roose‡

Report TW 252, January 1997

Department of Computer Science, K.U.Leuven

Abstract

We present a new numerical method for the efficient computation of periodic solutions of nonlinear systems of delay differential equations (DDEs) with several discrete delays. This method exploits the typical spectral properties of the monodromy matrix of a DDE and allows effective computation of the dominant Floquet multipliers to determine the stability of a periodic solution. We show that the method is particularly suited to trace a branch of periodic solutions using continuation and can be used to locate bifurcation points with good accuracy.

Keywords : Delay differential equations, periodic solutions, continuation, Newton-Picard.
AMS(MOS) Classification : Primary : 34K99, Secondary : 65C20.

*Accepted for publication in International Journal of Bifurcation and Chaos.
†Institute of Mathematical Problems in Biology, RAS, Pushchino, Moscow region, 142292, Russia
‡Department of Computer Science, K.U.Leuven, Celestijnenlaan 200A, B-3001 Leuven, Belgium

1 Introduction

The general theory and basic results for delay differential equations (DDEs) have been presented in the mathematical literature by many authors over the last years (see, e.g., [3, 11, 10, 15, 20, 16, 8]). However, many mathematical models with time delays cannot be treated by existing analytical methods or require complicated mathematical analysis. The investigation of such models requires numerical methods to analyse the stability and the bifurcation behaviour of the system under study. The development of these methods is difficult because of the specific properties of a dynamical system with delays. For this reason, research on numerical techniques for DDEs has mainly focussed until now on time integration (a bibliography is given in [1]).

An important topic of interest in many applications of DDEs is to show the existence of periodic solutions. There is an extensive literature on this topic. Authors have proved a number of fundamental results and they have suggested different methods for investigating the existence, stability and parameter dependence of periodic solutions for certain classes of DDEs [5, 25, 17, 18, 32, 26, 30, 6, 13, 21, 24]. However, these results and the corresponding methods are essentially theoretical in nature, have different rigorous restrictions, are well suited for some specific classes of DDEs but cannot be applied to a general nonlinear system with several delays.

Periodic solutions can arise via Hopf bifurcation of a steady state solution. So, the computation and bifurcation analysis of periodic solutions is not possible without some knowledge about Hopf bifurcation points. As the first step in the numerical bifurcation analysis of DDEs, we developed a numerical technique for the stability analysis of steady state solutions, the computation of Hopf bifurcation points and the continuation of branches of Hopf points for nonlinear systems of DDEs with several discrete delays [23]. The periodicity problem for DDEs is an infinite-dimensional problem because a system of DDEs is defined in an infinite-dimensional space. Indeed, computing a periodic solution of a DDE requires an initial function to be found, while only one initial point is required to characterize a periodic solution of a system of ODEs. Until now, not much purely numerical work has been done on this problem. All known approaches either use an approximation of the periodic solution by a Fourier series [9, 4, 31, 7], or use a shooting approach discretizing an initial function on the delay interval [14]. The first approach is quite efficient to trace branches of periodic solutions and even to compute homoclinic solutions [7]. However, it does not allow to determine the stability of the computed solutions. When applying the second approach, a discrete version of the Poincaré operator is constructed and all its eigenvalues, which are approximations of the Floquet multipliers of the periodic solution, are computed. Such an approach requires the solution of a large nonlinear system. Direct methods to deal with this are expensive and nonefficient, especially when we study a system of DDEs. Moreover, it is usually not necessary to compute a large number of Floquet multipliers to determine the stability of a periodic solution.

In this paper we present a new numerical method for the efficient computation of periodic solutions of DDEs and the determination of their stability. This approach exploits the fact that zero is a cluster point for the Floquet multipliers [15]. Usually only a few Floquet multipliers have large modulus. Therefore we can use the ideas behind the Newton-Picard method developed for the computation of periodic solutions of dissipative systems of PDEs [28, 22]. This algorithm can compute branches of periodic solutions and at little extra cost also the dominant, stability-determining Floquet multipliers. We now extend this method to compute periodic solutions of a DDE or a system of DDEs. We avoid an approximation of a DDE by a high order system of ODEs, which has been widespread in the mathematical and engineering literature [2]. The disadvantage of this approach is that an extremely large system of ODEs is necessary to obtain a good approximation for a periodic solution of a DDE, which leads to a very expensive method.

In Sec. 2 we formulate the periodicity problem for DDEs in an infinite dimensional space and suggest a finite dimensional approximation of this problem through the discretisation of an initial function on the delay interval. Using the discrete version of the Poincaré operator, we construct a nonlinear system which allows to compute the segment of the periodic solution on the delay interval and the value of its period. This leads to a single-shooting approach which is the starting point of the Newton-Picard algorithm. In Sec. 3 we briefly describe the main ideas of the Newton-Picard algorithm and consider its application to DDEs. Results of numerical experiments are presented in Sec. 4. In Sec. 5 we summarize the main results.

2 The Formulation of the Problem

We study the following system of DDEs with several discrete delays:

$$\frac{dx}{dt} = F(x(t), x(t-\tau_1), \ldots, x(t-\tau_m), \lambda), \tag{1}$$

where x is an n-dimensional vector of variables; τ_i, $i = 1, \ldots, m$, are the time delays ($\tau_i > 0$, $\tau = \max_i \tau_i$); λ is a parameter; F is a nonlinear n-dimensional vector-valued function. Let a solution of this system at time t be denoted by $x(t, \phi)$, where ϕ is an initial function, i.e. ϕ is a continuous function on $[-\tau, 0]$. Let $x_t(\phi)$ denote the segment of the solution defined by $x_t(\phi)(\theta) = x(t + \theta, \phi)$, $-\tau \leq \theta \leq 0$, or, in symbols, $x_t(\phi) \in C = C([-\tau, 0], I\!R^n)$, where C is the Banach space of continuous functions mapping the interval $[-\tau, 0]$ into $I\!R^n$. In order to simplify notations, we will now consider the case of a single DDE, but the following is also valid for a system of DDEs.

In studying the existence of a periodic solution of period T for a DDE we consider the Poincaré operator, which associates with each initial function ϕ defined on the interval $[-\tau, 0]$ the function $P\phi = x_T(\phi)$, i.e. the segment of the solution defined on $[T - \tau, T]$. So, to compute a periodic solution of a DDE, we need to find the solution of the fixed point problem $P\phi^* = \phi^*$, where the fixed point ϕ^* is a point in an infinite dimensional space.

For a periodic solution of (1) the linearized Poincaré operator S ($S = \partial x_T(\phi)/\partial\phi$) is defined by the equality

$$S\Delta\phi = y_T(\Delta\phi) \tag{2}$$

where $y_T(\Delta\phi)$ is the solution on $[T - \tau, T]$ of the variational equation

$$\frac{dy}{dt} = \sum_{i=0}^{m} f_i(x^*(t, \phi), x^*(t - \tau_1, \phi), \ldots, x^*(t - \tau_m, \phi))y(t - \tau_i) \tag{3}$$

with initial function $\Delta\phi$. Here $\tau_0 := 0$, f_i is the derivative of F with respect to the variable $x(t - \tau_i)$. This equation is obtained by linearizing the Eq. (1) along the solution $x^*(t, \phi)$ and it is also a delay equation.

The eigenvalues of the operator S are the Floquet multipliers of the periodic solution $x^*(t, \phi)$. It was proved [15] that the linearized Poincaré operator for differential equations with finite delays is compact and the Floquet multipliers are therefore eigenvalues of finite multiplicity with zero as their only cluster point. Furthermore, $\mu = 1$ is always a multiplier.

Thus, a periodic solution of (1) is determined by an initial function $\phi(\theta)$, $-\tau \leq \theta \leq 0$, and the period T. To find these unknowns, we use the system

$$\begin{cases} r(\phi, T) = x_T(\phi) - \phi = 0 \\ s(\phi, T) = 0. \end{cases} \tag{4}$$

The second equation is a phase condition needed to remove the indeterminacy due to the fact that the phase shift of any periodic solution is also a periodic solution. This means that periodic solutions are viewed as solutions of a BVP of the form (4).

The left-hand sides of (4) define a mapping of the space $C[-\tau,0]\times I\!R$ into $C[-\tau,0]\times I\!R$. When using Newton's method for solving (4), a linear system of the form

$$\begin{bmatrix} S^{(\nu)}-I & g^{(\nu)} \\ \frac{\partial s^{(\nu)}}{\partial\phi} & \frac{\partial s^{(\nu)}}{\partial T} \end{bmatrix} \begin{bmatrix} \Delta\phi^{(\nu)} \\ \Delta T^{(\nu)} \end{bmatrix} = -\begin{bmatrix} r(\phi^{(\nu)},T^{(\nu)}) \\ s(\phi^{(\nu)},T^{(\nu)}) \end{bmatrix} \tag{5}$$

must be solved in every iteration step $(^{\nu})$, where

$$S^{(\nu)} = S\big|_{(\phi^{(\nu)},T^{(\nu)})} = \frac{\partial x_T(\phi)}{\partial\phi}\bigg|_{(\phi^{(\nu)},T^{(\nu)})}, \quad g^{(\nu)} = \frac{\partial x_T(\phi)}{\partial T}\bigg|_{(\phi^{(\nu)},T^{(\nu)})} \in C[-\tau,0].$$

At the periodic solution S has an eigenvalue 1 with corresponding eigenfunction g:

$$Sg = g. \tag{6}$$

The new approximation for the periodic solution is computed via

$$\phi^{(\nu+1)} = \phi^{(\nu)} + \Delta\phi^{(\nu)}, \quad T^{(\nu+1)} = T^{(\nu)} + \Delta T^{(\nu)}, \quad \nu = 0,1,2,\dots \quad .$$

The first step of a numerical technique to compute a periodic solution (ϕ,T) is the discretisation of the function ϕ. We choose a meshsize $h=\tau/(N-1)$ and replace the function $\phi\in C[-\tau,0]$ by a vector of N values $\phi_i=\phi(-\tau+(i-1)h)$, $i=1,\dots,N$. Then we compute the solution of (1) by an integration code until $t=T$ and save values of the solution $x(t,\phi)$ on the interval $[T-\tau,T]$. After that, by a suitable interpolation, we compute approximate values of $x(t,\phi)$, $x_i=x(t_i,\phi)$, for the points $t_i=T-\tau+(i-1)h$, $i=1,\dots,N$. These N values x_i form a vector, which we take as an approximation of $x_T(\phi)$. Note that a nonuniform grid of points t_i in the intervals $[-\tau,0]$ and $[T-\tau,T]$ can be used also, as long as this grid is the same for both intervals.

The matrix of the discrete version of the operator S (which will be denoted by M) can be computed in different ways. We mention two standard procedures.

1) Using the variational equation: We compute the solution of (3) for $0\le t\le T$ with initial vectors e_j (the j-th unit vector) for $j=1,\dots,N$. Using an interpolation, we find the values of these solutions at points $t_i=T-\tau+(i-1)h$, $i=1,\dots,N$ and arrange these values columnwise in a square matrix M of order N.

2) Using finite differences: We compute the solution of (1), $\bar{x} = (x_1, \ldots, x_N)$, for the initial vector $\bar{\phi} = (\phi_1, \ldots, \phi_N)$. Then, starting with a perturbed initial vector $\tilde{\phi} = (\phi_1, \ldots, \phi_{j-1}, \tilde{\phi}_j = \phi_j + \epsilon, \phi_{j+1}, \ldots, \phi_N)$, we compute the solution of (1) $\tilde{x}^{(j)} = (\tilde{x}_1^{(j)}, \ldots, \tilde{x}_N^{(j)})$ and calculate the j-th column of M as $M_{i,j} = (\tilde{x}_i^{(j)} - x_i)/\epsilon$, $i = 1, \ldots, N$. Repeating this procedure N times ($j = 1, \ldots, N$), we compute M.

In both cases, we need to solve N initial value problems (IVP) of a delay equation (the variational equation or (1)) to compute the matrix M.

The discrete version of $g = \partial x_T / \partial T$ is formed during the integration of (1). As a phase condition $s(\phi, T) = 0$, we can, for instance, specify a Poincaré section:

$$\sum_{i=1}^{N} F_i^0 (\phi_i^0 - \phi_i) = 0, \tag{7}$$

where $(\phi_1^0, \ldots, \phi_N^0)$ is an initial vector, $(\phi_1, \ldots, \phi_N)$ is the solution vector, F_i^0 is the time derivative at point ϕ_i^0.

In this way, the linear operator equation (5) is replaced by a linear algebraic system of dimension $N + 1$

$$\begin{bmatrix} M^{(\nu)} - I & g^{(\nu)} \\ c^{(\nu)^T} & d^{(\nu)} \end{bmatrix} \begin{bmatrix} \Delta\phi^{(\nu)} \\ \Delta T^{(\nu)} \end{bmatrix} = - \begin{bmatrix} r(\phi^{(\nu)}, T^{(\nu)}) \\ s(\phi^{(\nu)}, T^{(\nu)}) \end{bmatrix}, \tag{8}$$

where

$$c^{(\nu)^T} = \left.\frac{\partial s}{\partial \phi}\right|_{(\phi^{(\nu)}, T^{(\nu)})}, \quad d^{(\nu)} = \left.\frac{\partial s}{\partial T}\right|_{(\phi^{(\nu)}, T^{(\nu)})}.$$

Note that in Eq. (8) and in the rest of this paper ϕ, x_T, g, c and r denote N-dimensional vectors. A straightforward method for computing a periodic solution is solving (8) by a direct linear system solver. At the periodic solution (ϕ^*, T^*) we call $M^* = M(\phi^*, T^*)$ the *monodromy matrix* — inspired by the terminology used in ODEs. The stability of the periodic solution can be determined by computing the eigenvalues of the monodromy matrix M^*. The discrete version of (6) holds only approximately: g^* is not an exact eigenvector of M^* and 1 is not an exact eigenvalue. However, the monodromy matrix M^* has an eigenvalue μ close to 1 with a corresponding eigenvector $\hat{g}$ close to g^*.

$$M^* \hat{g} = \mu \hat{g}, \quad g^* \approx \hat{g}, \quad \mu \approx 1. \tag{9}$$

The latter property can be used to check the accuracy of the calculated results (see Sec. 4).

It is clear that in this Newton procedure the construction of $M^{(\nu)}$ itself is computationally expensive, especially if we need to use a fine discretisation (large N) and/or when studying a system of DDEs.

3 Newton-Picard Methods

In the previous section, we introduced single shooting combined with Newton's method to compute periodic solutions of a system of DDEs. With this approach, one can compute both stable and unstable limit cycles. The method usually requires very few iterations. Moreover, we can also easily compute the Floquet multipliers. However, it has one serious disadvantage: the computational cost of each iteration step is extremely high when a fine discretisation is used on the delay interval due to the large monodromy matrix M that must be build.

When the limit cycle is strongly attracting, we can think of computing it by just integrating over a long time interval until we approach the limit cycle. By using a Poincaré section, one could also determine the period. This method is very cheap when the limit cycle is strongly attracting, but is slow for weakly stable and diverges for unstable limit cycles. Moreover, we do not obtain much information about the Floquet multipliers.

In our approach we exploit the fact that, for a system of DDEs, zero is a cluster point for the Floquet multipliers [15]. Usually, most Floquet multipliers are very close to zero and only a few Floquet multipliers lie close to or outside the unit circle. This can be exploited using the Newton-Picard method developed for the computation of periodic solutions of dissipative systems of PDEs [28, 22]. The method basically combines Newton-based single shooting for the weakly stable and unstable modes with time integration or a simple Picard scheme in the orthogonal complement. It avoids the high cost related to Newton's method, — the construction of the Jacobian matrix — and only uses selected matrix-vector products with the monodromy matrix. We will briefly present the main ideas of this approach. More detailed information can be found in [22].

3.1 Derivation of the Newton-Picard method

The Newton-Picard method relies on an appropriate splitting of the space in a low-dimensional subspace — the eigenspace corresponding to the Floquet multipliers larger than a threshold ρ — and its orthogonal complement. The dimension of the low-dimensional subspace follows from the following assumption:

Assumption 3.1 *Let $y^* = (\phi^*, T^*)$ denote an isolated solution to (4) after discretisation, and let $\mathcal{B}$ be a small neighbourhood of y^*. Let $M(y) = \frac{\partial x_T(\phi)}{\partial \phi}(y)$ for $y \in \mathcal{B}$ and denote its eigenvalues by μ_i, $i = 1, \ldots, N$. Assume that for all $y \in \mathcal{B}$ precisely p eigenvalues lie outside the disk*

$$C_\rho = \{|z| \leq \rho\}, \quad 0 < \rho < 1$$

and that no eigenvalue has modulus ρ, i.e. for all $y \in \mathcal{B}$

$$|\mu_1| \geq |\mu_2| \geq \cdots \geq |\mu_p| > \rho > |\mu_{p+1}| \geq \ldots \geq |\mu_N|.$$

Furthermore, ρ should be chosen small enough such that the eigenvalue of $M^* = M(y^*)$ corresponding to the trivial Floquet multiplier 1 of the linearised Poincaré operator (2) at the limit cycle is larger than ρ in modulus.

M^* is the monodromy matrix. Our method is designed to be efficient when $p \ll N$. Let $V_p \in \mathbb{R}^{N\times p}$ be a basis for the subspace $\mathcal{U}$ of $\mathbb{R}^N$ spanned by the (generalized) eigenvectors of $M(y)$ corresponding to the eigenvalues μ_i, $i = 1, \ldots, p$ and let $V_q \in \mathbb{R}^{N\times(N-p)} = \mathbb{R}^{N\times q}$ be a basis for $\mathcal{U}^\perp$, the orthogonal complement of $\mathcal{U}$ in $\mathbb{R}^N$. In general, $\mathcal{U}^\perp$ is not an invariant subspace of $M(y)$.

We can construct orthogonal projectors P and Q of $\mathbb{R}^N$ onto $\mathcal{U}$ and $\mathcal{U}^\perp$ respectively as

$$\begin{aligned} P &:= V_p V_p^T \\ Q &:= I - P = V_q V_q^T = I - V_p V_p^T. \end{aligned} \tag{10}$$

For any $\phi \in \mathbb{R}^N$ there is the unique decomposition

$$\phi = V_p\bar{\phi}_p + V_q\bar{\phi}_q = \phi_p + \phi_q, \quad \phi_p = V_p\bar{\phi}_p = P\phi, \quad \phi_q = V_q\bar{\phi}_q = Q\phi \tag{11}$$

with $\bar{\phi}_p \in \mathbb{R}^p$ and $\bar{\phi}_q \in \mathbb{R}^q = \mathbb{R}^{N-p}$. We decompose the Newton iteration (8) by inserting (11) and multiplying the first N columns at the left hand side with $\begin{bmatrix} V_q & V_p \end{bmatrix}^T$ and obtain

$$\begin{bmatrix} V_q^T(M-I)V_q & 0 & V_q^T g \\ V_p^T M V_q & V_p^T(M-I)V_p & V_p^T g \\ c^T V_q & c^T V_p & d \end{bmatrix} \begin{bmatrix} \Delta\bar{\phi}_q \\ \Delta\bar{\phi}_p \\ \Delta T \end{bmatrix} = - \begin{bmatrix} V_q^T r \\ V_p^T r \\ s \end{bmatrix} \tag{12}$$

For sake of clarity, we dropped the superscript (ν), and will continue to do so in all following equations. We have used $V_p^T V_q = 0_{p\times q}$, $V_q^T V_p = 0_{q\times p}$ and $V_q^T M V_p = 0_{q\times p}$ with the latter equality holding since $\mathcal{U}$ is an invariant subspace of $M(y)$. Because g is almost an eigenvector of M for an eigenvalue $> \rho$ (see (9)), and because the eigenvector corresponding to the latter eigenvalue is included in the subspace $\mathcal{U}$ by assumption 3.1, $V_q^T g \approx 0$ at the limit cycle and it is also very close to 0 near the limit cycle. Therefore we can neglect this term, and so we end up with a block lower triangular system. The upper left block is still a high-dimensional system. The matrix $V_q^T(M-I)V_q$ cannot be constructed explicitly at a reasonable cost. However, because of the construction of the basis, the spectral radius of $V_q^T M V_q$, $r_\sigma(V_q^T M V_q) = |\mu_{p+1}| < \rho < 1$. Therefore, we can solve these equations approximately using the Picard scheme

$$\Delta\bar{\phi}_q^{[i+1]} = V_q^T M V_q \Delta\bar{\phi}_q^{[i]} + V_q^T r \tag{13}$$

with starting value $\Delta\bar{\phi}_q^{[0]} = 0$. The first step of this Picard scheme is just the projection of the time integration on the basis V_q. To avoid the need for the basis V_q, we immediately compute the N-dimensional update $\Delta\phi_q = V_q\Delta\bar{\phi}_q$ by means of the Picard iteration

$$\Delta\phi_q^{[i+1]} = QM\Delta\phi_q^{[i]} + Qr. \tag{14}$$

(14) is obtained from (13) by premultiplying (13) with V_q and using the definition of the projectors (10). We perform a fixed and user-determined number of Picard steps l and take $\Delta\phi_q = \Delta\phi_q^{[l]}$. Remark that l steps of iteration scheme (13) correspond to approximating the inverse of $V_q^T(M-I)V_q$ with the first l terms of the Neumann series

$$(V_q^T(M-I)V_q)^{-1} \approx -\sum_{i=0}^{l-1}(V_q^T M V_q)^i.$$

One or two steps of the Picard scheme (14) are generally enough to obtain overall convergence of the Newton-Picard method. Once $\Delta\phi_q$ has been computed, we can solve $\Delta\bar{\phi}_p$ and ΔT from

$$\begin{bmatrix} V_p^T(M-I)V_p & V_p^T g \\ c^T V_p & d \end{bmatrix} \begin{bmatrix} \Delta\bar{\phi}_p \\ \Delta T \end{bmatrix} = - \begin{bmatrix} V_p^T(r + M\Delta\phi_q) \\ s + c^T\Delta\phi_q \end{bmatrix}. \tag{15}$$

This is only a $(p+1)\times(p+1)$ system that can be constructed cheaply. In fact, the matrix-vector products MV_p have already been computed during the subspace iterations as is shown in [22]. The system is solved using Gauss elimination or a least squares approach based on the generalized inverse. The latter offers advantages when the matrix of (15) is close to singular.

Remark that performing one step of the Picard scheme (14) corresponds to performing one time integration around the limit cycle in $\mathcal{U}^\perp$:

$$\phi_q^{(\nu+1)} = Qx_{T^{(\nu)}}(\phi_p^{(\nu)} + \phi_q^{(\nu)}) = Qx_{T^{(\nu)}}(\phi^{(\nu)}). \tag{16}$$

Performing multiple Picard steps corresponds to multiple time integrations up to higher order terms. System (15) is equivalent to

$$\begin{aligned}\begin{bmatrix} V_p^T(M-I)V_p & V_p^T g \\ c^T V_p & d \end{bmatrix}\begin{bmatrix} \Delta\bar{\phi}_p^{(\nu)} \\ \Delta T^{(\nu)} \end{bmatrix} &= -\begin{bmatrix} V_p^T r(\phi_p^{(\nu)} + \phi_q^{(\nu+1)}, T^{(\nu)}) \\ s(\phi_p^{(\nu)} + \phi_q^{(\nu+1)}, T^{(\nu)}) \end{bmatrix} + \text{h.o.t.} \\ &= -\begin{bmatrix} V_p^T r(\phi^{(\nu)} + \Delta\phi_q^{(\nu)}, T^{(\nu)}) \\ s(\phi^{(\nu)} + \Delta\phi_q^{(\nu)}, T^{(\nu)}) \end{bmatrix} + \text{h.o.t.}\end{aligned} \tag{17}$$

(16) and (17) clearly show that basically, our method first performs one or more time integrations in the subspace $\mathcal{U}^\perp$, followed by a Newton single shooting step in the subspace $\mathcal{U}$.

3.2 Computational aspects

To show that this procedure can be executed efficiently, it remains to show how to compute the basis and how to do the matrix-vector multiplications with the monodromy matrix M without completely constructing that matrix.

To compute the basis, we use subspace iteration with projection and locking [29, 22]. However, some modification to the algorithm presented in [22] have been made to make the overall method more robust. We do not only compute the basis V_p, but add some extra vectors to improve the convergence of the more dominant vectors in the subspace iteration and to make sure that all eigenvectors corresponding to eigenvalues larger than ρ are included in the basis. Note that both p and the number of extra vectors may vary along the computed branch of periodic solutions and we need strategies to cope with this. A precise discussion of these strategies is outside the scope of this article. The implementation used for the numerical tests performs one or more subspace iteration steps between each Newton-Picard step until at least p basis vectors are accurately enough. No locking is performed between two Newton-Picard steps, but locking of accurate vectors is used when more than one subspace iteration step is performed before the Newton-Picard step. If more than p basis vectors are accurately enough for the iterations, all accurate vectors are used, which corresponds to lowering the actual value of ρ for that step.

It remains to show how a matrix-vector product Mv can be computed efficiently. This can be done using variational equations or finite differences. In the first approach, we start from the fundamental matrix $W(t)$ which is the solution of

$$\dot{W}(t) = \sum_{i=0}^{m} f_i(x(t,\phi), x(t-\tau_1,\phi), \ldots, x(t-\tau_m,\phi))W(t-\tau_i). \tag{18}$$

These equations are called the variational equations. They are discretised on the delay interval using a finite dimensional approximation as described in Sec. 2. $W(t)$ is then an $N\times N$-matrix. The discretised version of equation (18) is an N-dimensional linear initial-value problem with delays if $x(t,\phi)$ is known. At the solution $x^*(t,\phi)$, $M^* = W(T)$. In our algorithm, we only need to compute Mv for a certain number of vectors $v \in \mathbb{R}^N$. These can be computed from the discretised version of

$$\dot{w}(t) = \sum_{i=0}^{m} f_i(x(t,\phi), x(t-\tau_1,\phi), \ldots, x(t-\tau_m,\phi))w(t-\tau_i) \tag{19}$$

where the initial data on the delay interval $[-\tau, 0]$ is taken from the vector v. (19) is obtained from (18) by right-multiplying with v and putting $w(t) = W(t)v$. Using first-order accurate forward differences, an approximation to

$$Mv = \left.\frac{\partial x_T(\phi)}{\partial \phi}\right|_{(\phi,T)} v$$

is given by

$$Mv \approx \frac{1}{\varepsilon}[x_T(\phi+\varepsilon v) - x_T(\phi)]. \tag{20}$$

Thus in both approaches, the calculation of the action of M on a vector v requires the solution of one initial value problem with delays (at least if $x(t,\phi)$ or $x_T(\phi)$ is known).

Similarly, to calculate the action of M on the p-dimensional subspace $\mathcal{U}$, p initial value problems need to be solved.

To conclude, we give the skeleton of the overall algorithm. To avoid obscuring the main ideas, the algorithm is given without projection or locking for the subspace iteration steps, although we did use projection and locking to obtain the test results.

Algorithm 3.1 *Newton-Picard with subspace iteration, l Picard steps*

Input: Starting value ϕ, T and the starting basis $V_e = [v_1 \cdots v_p \cdots v_{p_e}]$
Output: FAILURE or ϕ^*, T^* and the final basis V_e.
begin
  $\nu = 0$
  $W_e \leftarrow V_e$
  repeat
    Compute $x_T(\phi)$, $r = x_T(\phi) - \phi$.
    while $[v_1 \cdots v_p]$ not accurately enough **do**
      Perform a subspace iteration step : $V_e \leftarrow \text{ortho}(W_e)$, $W_e \leftarrow MV_e$.
      Check whether vectors have to be added to the basis.
    Choose the effective value of p that will be used for the Newton-Picard step.
    Select V_p, $W_p = MV_p$.
    $\Delta\phi_q \leftarrow 0$
    for $i = 1$ **to** l **do** $\Delta\phi_q \leftarrow (I - V_p V_p^T)(M\Delta\phi_q + r)$
    Compute the Newton correction from

$$\begin{bmatrix} V_p^T W_p - I_p & V_p^T g \\ c^T V_p & d \end{bmatrix} \begin{bmatrix} \Delta\bar{\phi}_p \\ \Delta T \end{bmatrix} = - \begin{bmatrix} V_p^T(r + M\Delta\phi_q) \\ s + c^T \Delta\phi_q \end{bmatrix}.$$

    Update the point:

$$\phi \leftarrow \phi + \Delta\phi_q + V_p \Delta\bar{\phi}_p, \;\; T \leftarrow T + \Delta T.$$

    Check whether vectors have to be removed from the basis.
    $\nu \leftarrow \nu + 1$
  until convergence **or** $\nu \geq \nu_{\max}$
  if no convergence **then return** FAILURE
  else $\phi^* \leftarrow \phi$, $T^* \leftarrow T$.
end

The above algorithm requires one evaluation of the residual r per iteration step and at least $p_e + l$ matrix-vector products with M: p_e matrix-vector products during the subspace iteration steps, $l-1$ during the Picard iterations and one to build the right-hand side of the Newton system. In the beginning, more than one subspace iteration step may be needed, but as the Newton-Picard procedure converges, one subspace iteration step is enough to ensure an accurate enough basis.

3.3 Continuation of periodic orbits

The approach presented in Sec. 3.1 can be used in a continuation code. A predictor is used to construct a first approximation for the next periodic solution on the branch, and afterwards, the Newton-Picard procedure is used for the corrector iterations that move the point onto the branch. However, the corrector computes a periodic solution at the same parameter value as the predicted point. Therefore, this approach can fail in the neighbourhood of a fold point. In a continuation context, this problem is typically solved by making the continuation parameter free during the corrector iteration and by adding another equation to have a fully determined system. We follow the pseudo-arclength approach proposed by Keller [19] and solve the extended system

$$\left\{ \begin{array}{l} r(\phi, T, \lambda) = x_T(\phi; \lambda) - \phi = 0 \\ s(\phi, T, \lambda) = 0 \\ u(\phi, T, \lambda; \sigma) = 0. \end{array} \right. \tag{21}$$

Here λ is an unknown parameter and σ is an artificial parameter which determines the steplength along the curve to be traced. With a careful choice of the arclength condition $u(\phi, T, \lambda; \sigma) = 0$, the system remains regular at fold points which can be passed without any difficulty. However, whatever choice is used, the system is still singular at certain higher codimension bifurcation points such as transcritical or pitchfork bifurcations.

After projection of the Newton linearisation of (21), we obtain a system similar to (12). However, this system is not block triangular anymore. We solve it using the approach presented in [12]. We have also implemented a version of the algorithm that does not explicitly use the steplength condition $u = 0$, but instead solves the low-dimensional underdetermined system using the generalized inverse. This procedure implicitly defines a good steplength condition and gives much better results than Gauss elimination and the explicit pseudo-arclength condition in the neighbourhood of transcritical or pitchfork bifurcations.

3.4 Remarks

The computational cost of the Newton-Picard algorithm is dominated by the integration of the nonlinear system (1) needed to compute the residual r and the matrix-vector products with the monodromy matrix. The cost of the other operations is almost negligible and proportional with N. As is shown in [22], the convergence behaviour of the method is fully determined by the dominant Floquet multipliers. The improvement of the residual obtained by one Newton-Picard step and the amount of matrix-vector products for one step are almost independent of the time discretisation on the delay interval, as long as the limit cycle and its dominant Floquet multipliers are represented well enough. Compared with this, the computational cost of one Newton step grows much faster as the time discretisation is refined, since the size of the monodromy matrix (and thus the number of columns that has to be computed, which requires essentially one matrix-vector product per column) grows linearly with N. Remark that a rather low accuracy of the time integration (and thus a small value of N) is often enough to compute a reasonable approximation to the periodic solution away from bifurcation points, but is often not enough to obtain a good accuracy for the Floquet multipliers. These should be computed with enough accuracy to detect bifurcation points correctly.

It is also important to note that our method has no restrictions on the dimension of the system nor on the number or the values of the delays. Some other methods are restricted to a system with commensurable delays (e.g., [15]) or to a system with one delay [7].

4 Numerical Results

In this section we present the results of the application of the Newton-Picard approach for computing periodic solutions of delay differential equations. Time integration of (1) was implemented using a fourth-order Runge-Kutta method with adaptive step size and a fourth-order interpolation scheme. Matrix-vector products with the monodromy matrix were implemented using finite differences (20).

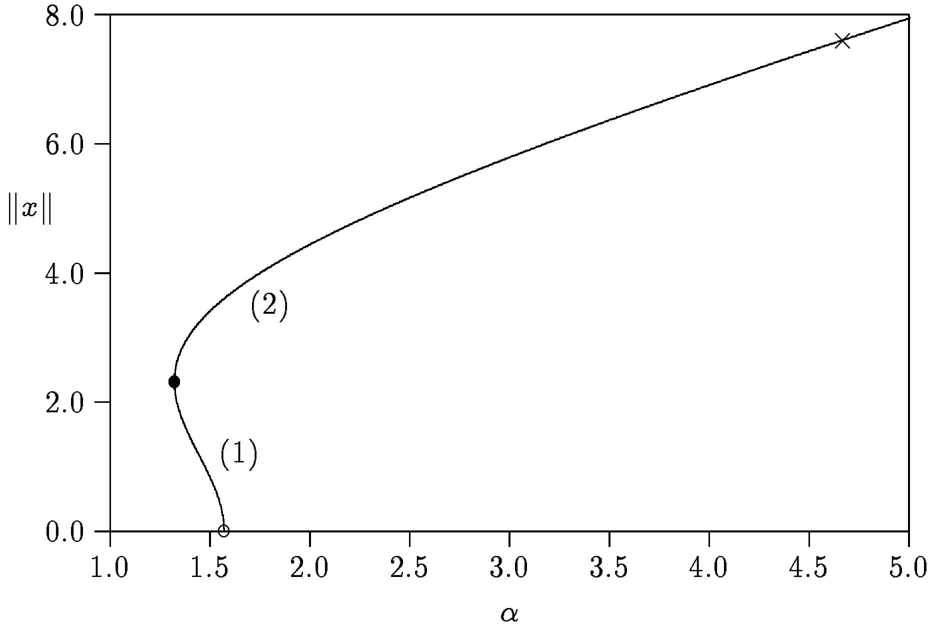

Figure 1: The norm of the solution on the delay interval along a branch of periodic solutions of (22). ○ - Hopf bifurcation, ● - turning point, × - transcritical bifurcation.

4.1 Example 1

As a first example we choose an equation that was studied analytically (see, e.g., [17, 26, 32]) and numerically [14, 9]:

$$\frac{dx}{dt} = -\alpha x(t-1)\frac{1 + x^2(t-1)}{1 + x^4(t-1)}. \tag{22}$$

It was proved that at $\alpha = \pi/2$ a family of periodic solutions bifurcates from the zero solution, solutions of period $T = 4$ exist for $\alpha > \pi/2$ and that a fold bifurcation (a turning point) exists for α close to $\pi/2$.

We computed a branch of periodic solutions with the Newton-Picard method using $N = 21$ and $\rho = 0.3$. Along this branch no more than two Floquet multipliers are larger

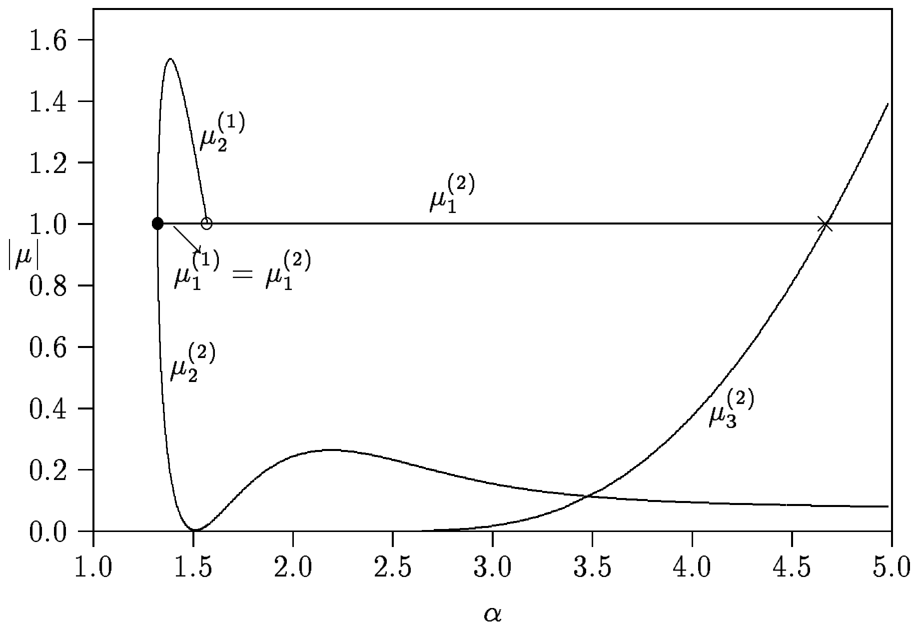

Figure 2: The dominant Floquet multipliers along the branch of periodic solutions of (22), shown in Fig. 1. $\mu_1^{(1)}, \mu_2^{(1)}$ denote Floquet multipliers of the lower part of the branch (below the turning point), $\mu_1^{(2)}, \mu_2^{(2)}, \mu_3^{(2)}$ denote Floquet multipliers of the upper part of the branch.

than ρ in modulus. Hence, in the algorithm of Sec. 3.2, $p = 2$ would be sufficient, but for safety reasons 2 extra multipliers were computed within the subspace iteration. Figures 1 and 2 show the norm of the computed periodic solution on the delay interval and the behaviour of the moduli of the dominant Floquet multipliers, respectively, versus the parameter α. We detected three bifurcation points along this branch: a Hopf bifurcation point ($\alpha \approx 1.57$), a turning point ($\alpha \approx 1.32$) and a transcritical bifurcation point ($\alpha \approx 4.67$). Note that the Floquet multipliers $\mu_4, \ldots, \mu_{21}$ remain smaller than 0.0044 along the branch.

In order to check the convergence and the accuracy of the obtained results, we computed a point on the branch of periodic solutions for different values of N: $N = 11; 21; 41; 81$. Table 1 shows the values of the first 4 Floquet multipliers and the period for this point. Because for a periodic solution one of the Floquet multipliers is always 1 (6), we can exploit this property to check the accuracy of the calculated results. We note that there is no significant difference between the values of the multipliers for $N = 41$ and $N = 81$.

	$\lvert\mu_1\rvert$	$\lvert\mu_2\rvert$	$\lvert\mu_3\rvert$	$\lvert\mu_4\rvert$	T
$N = 11$	1.00695464	0.76687110	0.08439711	0.00271824	3.9986426
$N = 21$	0.99999327	0.77952649	0.08396869	0.00302678	3.9999036
$N = 41$	0.99999556	0.77999131	0.08395139	0.00291003	3.9999935
$N = 81$	0.99999958	0.78002082	0.08394678	0.00287782	3.9999996

Table 1: Moduli of the first 4 Floquet multipliers and the period for a point on the branch of periodic solutions of (22) ($\alpha = 4.4745$).

Table 2 confirms our considerations of the end of the last section. It presents the number of IVP solves per continuation step for 12 points computed along a segment of the branch ($3.37 \leq \alpha \leq 4.38$) using a fixed continuation step size. We see that the number of IVP solves generally does not depend on N (except for the case $N = 11$ when the number of IVP solves grows because of the bad approximation). So, using the technique described above, we can get a good accuracy for the dominant Floquet multipliers by increasing the value of N, which does not lead to an increase of the number of IVP solves. Note that for $N = 81$, the construction of the Jacobian matrix

	1	2	3	4	5	6	7	8	9	10	11	12	total
$N = 11$	86	73	56	58	64	86	109	84	90	88	82	78	854
$N = 21$	78	77	53	61	53	55	49	61	68	66	63	60	744
$N = 41$	62	61	67	60	60	55	55	59	55	60	60	57	721
$N = 81$	62	69	69	63	59	55	55	55	59	60	57	57	720

Table 2: The number of time integrations per continuation step for 12 points on the branch of periodic solutions of (22) ($\alpha \in [3.37, 4.38]$).

in one full Newton iteration step would require 81 IVP solves.

4.2 Example 2

To obtain a second example we started from the Olmstead model [27], which describes the flow of a viscoelastic fluid (i.e. a fluid with a “fading memory”):

$$\frac{du}{dt} = \int_{-\infty}^{t} \frac{1-\delta}{\nu} e^{-(t-s)/\nu} u_{xx}(x,s)ds + \delta u_{xx}(x,t) + Ru(x,t) - u^3(x,t) \qquad (23)$$

on $x \in I = (0, \pi)$ with boundary conditions $u(0,t) = u(\pi,t) = 0$. Here $u(x,t)$ is the velocity of the fluid, ν, δ, R are parameters of the problem.

The classical method to study the integro-differential equation (23) is the conversion of (23) into a pair of coupled partial differential equations without the integral (which represents a continuous delay). However, a "discrete" version of the Olmstead model can be obtained by approximating the integral in (23) using Simpson's rule.

$$\begin{aligned} \frac{du}{dt} = \frac{1-\delta}{\nu}&(w_0 u_{xx}(x,t) + w_1 u_{xx}(x,t-\tau_1) + w_2 u_{xx}(x,t-\tau_2)) \qquad (24) \\ &+ \delta u_{xx}(x,t) + Ru(x,t) - u^3(x,t) \end{aligned}$$

where τ_1, τ_2 are time delays and w_0, w_1, w_2 are weight coefficients of the quadrature formula.

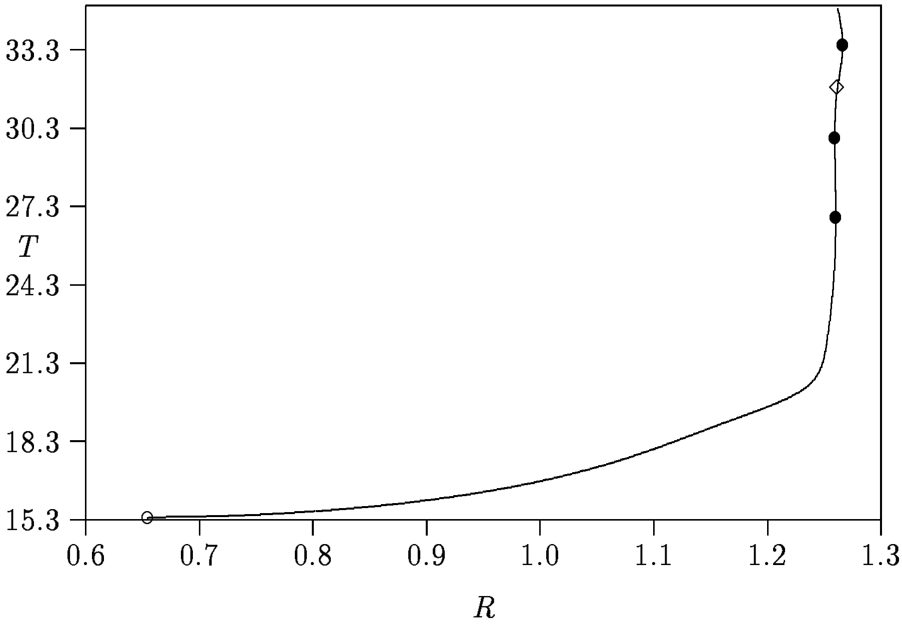

Figure 3: The period along a branch of periodic solutions of (25). ○ - Hopf bifurcation, ● - turning points, ◇ - torus bifurcation.

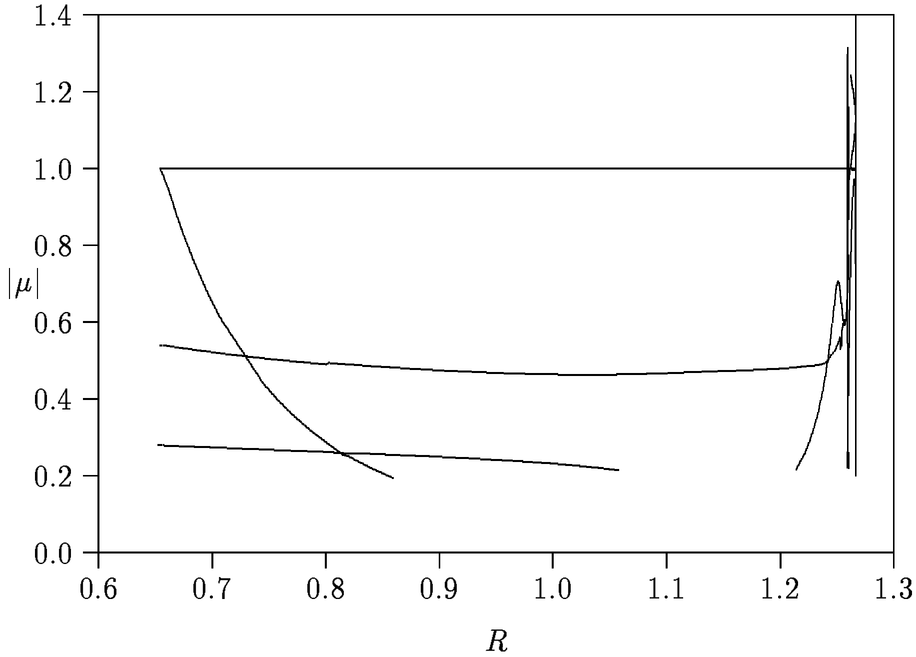

Figure 4: The dominant Floquet multipliers along the branch of periodic solutions of (25), shown in Fig. 3.

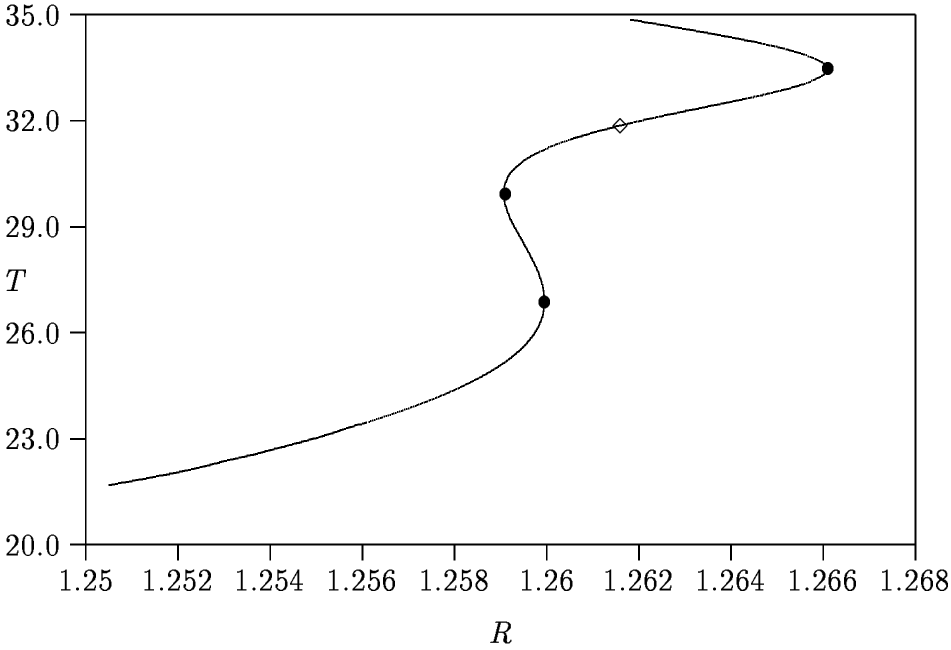

Figure 5: The period along the branch of periodic solutions of (25). ∘ - Hopf bifurcation, • - turning points, ⋄ - torus bifurcation.

Our aim here is not the investigation of (23) as a model of real physical phenomena. Equation (24) is only a suitable mathematical equation to test the numerical techniques described above and to compare results with known results for the model (23) presented e.g. in [27]. We use the classical second order central difference scheme in space to approximate (24) by a system of 8 delay equations:

$$\begin{aligned}\frac{du_i}{dt} = \frac{1-\delta}{\nu}\sum_{j=0}^{2} w_j f(u_{i-1}(s_j), u_i(s_j), u_{i+1}(s_j)) \\ + \delta f(u_{i-1}(t), u_i(t), u_{i+1}(t)) + Ru_i(t) - u_i^3(t), \quad i = 1, \ldots, n\end{aligned} \tag{25}$$

where $s_j = t - \tau_j$, $\tau_0 = 0$, $f(v_1, v_2, v_3) = (v_1 - 2v_2 + v_3)/h^2$, $u_0(t) = u_{n+1}(t) = 0$, $h = \pi/(n+1)$, $n = 8$.

For this system, branches of Hopf bifurcation points were computed and some regions of the stability of periodic solutions were obtained in [23]. Now we present results on computing periodic solutions of (25). We use R as an active parameter and fix the other parameters at the following values: $\delta = 0.2$, $\nu = 2$, $w_0 = 1$, $w_1 = 0.92$, $w_2 = 0.08$, $\tau_1 = 3$, $\tau_2 = 6$. A branch of periodic solutions was computed using $N = 21$ and $\rho = 0.2$. Figures 3 and 4 show the period and the behaviour of the computed Floquet multipliers, respectively, versus R. A maximum of 10 multipliers were calculated along the branch. We detected five bifurcation points: a Hopf bifurcation point ($R \approx 0.6544$), three turning points ($R \approx 1.2591; 1.2599; 1.2661$) and a torus bifurcation point ($R \approx 1.2616$). An enlargement of the last part of the branch is shown in Figs. 5-6. The vertical line in Fig. 6 around $R \approx 1.266$ is a Floquet multiplier that steeply ascends to numbers higher than 110 after which the accuracy of the computed Floquet multiplier 1 decreases drastically which is caused by the ill-conditioning of this Floquet multiplier. We conclude that even for reasonable values of N we are able to compute very unstable periodic solutions with large period ($T > 34$).

Figure 7 shows the moduli of the computed Floquet multipliers versus an approximation of the arclength of the branch. While small values of N might be good enough to calculate a branch of periodic solutions and its bifurcation behaviour, large values of N are needed when we want accurate computation of the Floquet multipliers especially in the neighbourhood of bifurcation points. In Fig. 7 we see how this loss of accuracy results in the forming of a complex pair of the Floquet multiplier 1 and the Floquet multiplier crossing through 1 in the neighbourhood of the first turning point. We computed the first turning point and the torus bifurcation point using a code that adapts the steplength of the continuation algorithm to approximate the bifurcation point. The results of these calculations are given in Table 3 and 4. We see that the presence of a second multiplier near 1 greatly diminishes the accuracy. However for both cases there is nice convergence for the parameter and the period.

Table 5 gives the moduli of the first 6 Floquet multipliers and the value of the period for a point on the curve using $N = 11; 21; 41$. Since (25) consist of 8 delay equations, the size of the monodromy matrix in the Newton-Picard algorithm is 88, 168 and 328, respectively. All computed quantities show good convergence. The number of time integrations per continuation step for 12 points on the branch of periodic solutions,

	$\lvert\mu_1 - 1\rvert = \lvert\mu_2 - 1\rvert$	R	T
$N = 11$	7.57e-02	1.259968795	26.8602059
$N = 21$	2.05e-02	1.259948241	26.8817127
$N = 41$	7.45e-03	1.259947692	26.8885988
$N = 81$	1.55e-03	1.259947847	26.8885441

Table 3: Approximation of a turning point on the branch of periodic solutions of (25) for different values of N.

	$\lvert\mu_1 - 1\rvert$	$\lvert\lvert\mu_2\rvert - 1\rvert = \lvert\lvert\mu_3\rvert - 1\rvert$	R	T
$N = 11$	1.20e-04	5.11e-09	1.261588171	31.858250
$N = 21$	1.58e-05	2.77e-07	1.261559955	31.850110
$N = 41$	3.33e-05	3.68e-08	1.261557901	31.849493
$N = 81$	1.39e-05	2.52e-07	1.261557024	31.849168

Table 4: Approximation of a torus bifurcation point on the branch of periodic solutions of (25) for different values of N.

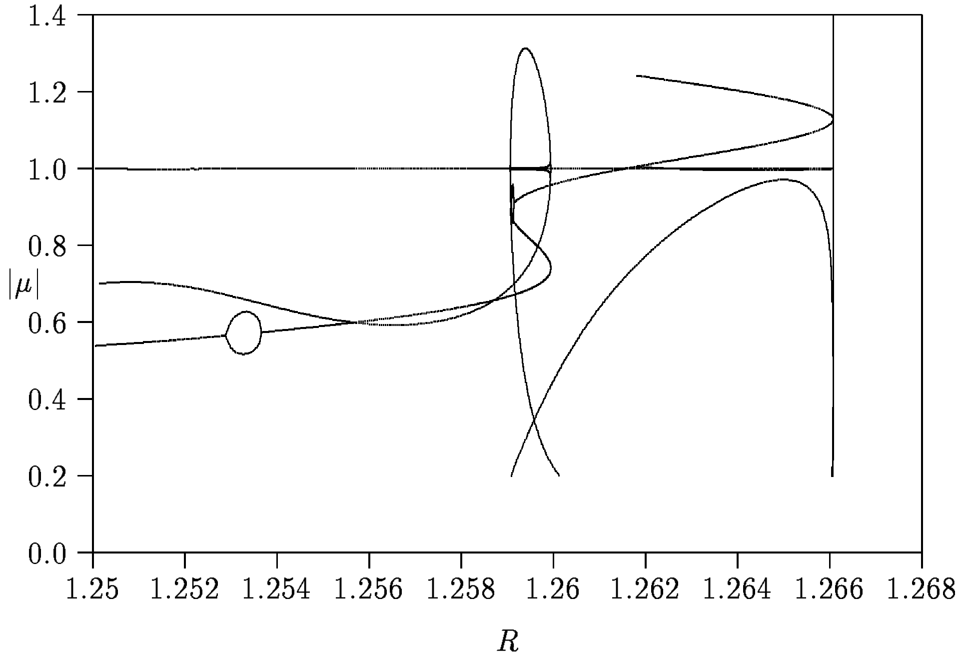

Figure 6: The dominant Floquet multipliers along the branch of periodic solutions of (25).

	$\lvert\mu_1\rvert$	$\lvert\mu_2\rvert = \lvert\mu_3\rvert$	$\lvert\mu_4\rvert$	$\lvert\mu_5\rvert = \lvert\mu_6\rvert$	T
$N = 11$	1.00047136	0.49008132	0.28830116	0.26154925	15.609848
$N = 21$	0.99998339	0.49035024	0.28834113	0.26183869	15.609690
$N = 41$	0.99999923	0.49037802	0.28834387	0.26186098	15.609658

Table 5: Moduli of the first 6 Floquet multipliers and the period for a point on the branch of periodic solutions of (25) ($R = 0.8$).

	1	2	3	4	5	6	7	8	9	10	11	12	total
$N = 11$	88	88	88	88	88	88	88	88	88	88	88	124	1138
$N = 21$	75	85	85	85	85	85	85	85	75	85	85	85	1000
$N = 41$	81	81	81	81	81	81	81	89	89	95	95	100	1035

Table 6: The number of time integrations per continuation step for 12 points on the branch of periodic solutions of (25) ($R \in [0.820, 0.967]$).

presented in Table 6, confirms the results of the first example: an increase of N does not lead to an increase of the number of IVP solves. The independence of the number of time integrations per Newton-Picard iteration on the accuracy of the discretisation (on the value of N) allows to use large values of N to compute dominant Floquet multipliers with good accuracy at a small extra computational cost. However, a very large value of N (a very fine discretisation) can increase the computational cost because the accuracy of the time integrator should be raised accordingly. While computing branches of periodic solutions we did not detect a significant difference in the computational time between the cases $N = 11$ and $N = 41$.

5 Conclusion

We have presented a new numerical method for the efficient computation of periodic solutions of nonlinear systems of delay differential equations (DDEs) with several discrete delays. This method exploits the typical spectral property of the monodromy matrix of

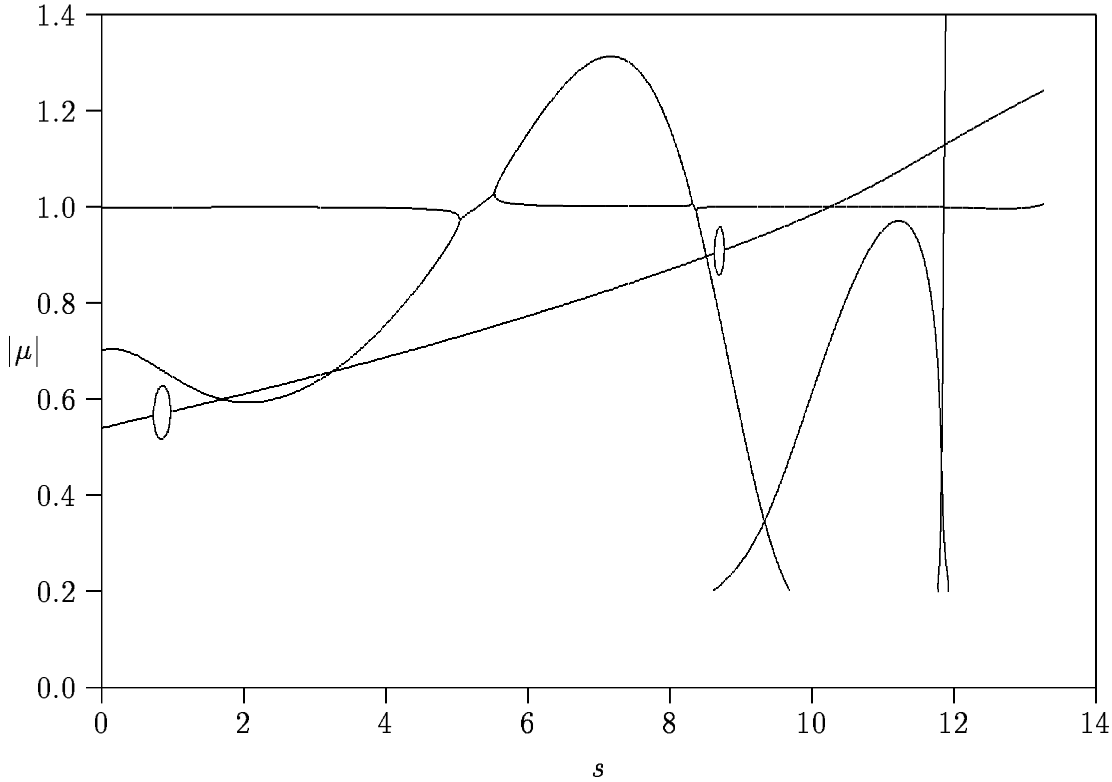

Figure 7: The dominant Floquet multipliers versus the arclength s of the branch of periodic solutions of (25).

a DDE and allows effective computation of the dominant Floquet multipliers to determine the stability of the computed periodic solution. The basic computational cost of the technique is determined by the number of time integrations. We have shown that the accuracy of the discretisation of the periodicity problem of a DDE has no effect on this number. This allows to compute periodic solutions and dominant Floquet multipliers with a high accuracy at a reasonable computational cost. Numerical experiments have also shown that the presented technique is quite efficient to trace a branch of periodic solutions using continuation and to locate bifurcation points along this branch with good accuracy.

Acknowledgements

This research is partially funded by the projects NFWO-G.0235.96, OT/94/16 (Research fund K.U.Leuven), the Belgian programme on Interuniversity Poles of Attraction (IUAP 17), initiated by the Belgian State — Prime Minister's Service — DWTC, Federal Office for Science, Technical and Cultural Affairs (OSTC). The scientific responsibility is assumed by its authors. T. Luzyanina has been funded by a fellowship of OSTC. K. Engelborghs and K. Lust are Research Assistants of the National Fund for Scientific Research (Belgium).

References

[1] C. T. H. Baker, C. A. H. Paul, and D. R. Willé. A bibliography on the numerical solution of delay differential equations. Technical Report 269, University of Manchester, Manchester Centre for Computational Mathematics, 1995.

[2] H. T. Bank and J. A. Burns. Hereditary control problems: numerical methods based on averaging approximations. *SIAM J. Control and Optim.*, 16(2):169–209, 1978.

[3] R. Bellman and K. L. Cooke. *Differential-Difference Equations*, volume 6 of *Mathematics in science and engineering*. Academic Press, 1963.

[4] A. M. Castelfranco and H. W. Stech. Periodic solutions in a model of recurrent neural feedback. *SIAM J. Appl. Math.*, 47(3):573–588, 1987.

[5] S.-N. Chow. Existence of periodic solutions of autonomous functional differential equations. *J. Diff. Eqns.*, 15:350–378, 1974.

[6] S.-N. Chow and H.-O. Walther. Characteristic multipliers and stability of symmetric periodic solutions of $x'(t) = g(x(t-1))$. *Trans. Amer. Math. Soc.*, 370(1):127–142, 1988.

[7] L. Cometto, M. Dahmen, and F. Giannakopoulos. On the number of p-zeroes of a quasi-polynomial with applications to the stability and Hopf-Bifurcation of systems of differential equations with delays. In preparation, 1996.

[8] O. Diekmann, S. A. van Gils, S. M. Verduyn Lunel, and O.-H.Walter. *Delay Equations: Functional-, Complex-, and Nonlinear Analysis*, volume 110 of *Applied Mathematical Sciences*. Springer-Verlag, 1995.

[9] E. J. Doedel and P. P. C. Leung. A numerical technique for bifurcation problems in delay differential equations. *Congr. Num.*, 34:225–237, 1982.

[10] R. D. Driver. *Ordinary and Delay Differential Equations*, volume 20 of *Applied Mathematical Science*. Springer-Verlag, 1977.

[11] L. E. El'sgol'ts and S. B. Norkin. *Introduction to the Theory and Application of Differential Equations with Deviating Arguments*, volume 105 of *Mathematics in science and engineering*. Academic Press, 1973.

[12] K. Engelborghs, K. Lust, and D. Roose. A Newton-Picard method for accurate computation of period doubling bifurcation points of large-scale systems of ODEs. TW report 251, Department of Computer Science, Katholieke Universiteit Leuven, Belgium, 1996.

[13] L. Erbe and X. Liu. Monotone iterative methods for differential systems with finite delay. *Appl. Math. Comp.*, 43:43–64, 1991.

[14] K. P. Hadeler. Effective computation of periodic orbits and bifurcation diagrams in delay equations. *Numer.Math.*, 34:457–467, 1980.

[15] J. K. Hale. *Theory of Functional Differential Equations*, volume 3 of *Applied Mathematical Science*. Springer-Verlag, 1977.

[16] J. K. Hale and S. M. Verduyn Lunel. *Introduction to Functional Differential Equations*. Springer-Verlag, 1993.

[17] J. Kaplan and J. Yorke. Ordinary differential equations which yield periodic solutions of differential-delay equations. *J. Math. Anal. Appl.*, 48:317–324, 1974.

[18] J. Kaplan and J. Yorke. On the stability of a periodic solution of a DDE. *SIAM J. Math. Anal.*, 6(2):268–282, 1975.

[19] Herbert B. Keller. Numerical solution of bifurcation and nonlinear eigenvalue problems. In P. H. Rabinowitz, editor, *Applications of Bifurcation Theory*, pages 359–385. Academic Press New York, 1977.

[20] V. B. Kolmanovskii and A.Myshkis. *Applied Theory of Functional Differential Equations*, volume 85 of *Mathematics and its Applications.* Kluwer Academic Publishers, Dordrecht, 1992.

[21] X. Liu. Periodic baundary value problem for delay equations with finite delay. *Dynamic Systems and Applications*, 3:357–368, 1994.

[22] K. Lust, D. Roose, A. Spence, and A. R. Champneys. An adaptive Newton-Picard algorithm with subspace iteration for computing periodic solutions. *SIAM J. Scientific Computing*, 1996. Accepted.

[23] T. Luzyanina and D. Roose. Numerical stability analysis and computation of Hopf bifurcation points for delay differential equations. *J. Comp. Appl. Math.*, 72:379–392, 1996.

[24] J. L. Moiola, H. G. Chiacchiarini, and A. C. Desages. Bifurcations and Hopf degeneracies in nonlinear feedback systems with time delay. *Int. J. Bifurcation and Chaos*, 6(4):661–672, 1996.

[25] R. D. Nussbaum. Periodic solutions of some nonlinear autonomous functional differential equations. *I. Ann. Mat. Pura Appl.*, 101:236–306, 1974.

[26] R. D. Nussbaum. Periodic solutions of nonlinear autonomous functional differential equations. In H.-O. Peitgen and H.-O. Walther, editors, *Functional Differential Equations and Approximation of Fixed Points*, volume 730 of *Lecture Notes in Mathematics*, pages 283–325. Springer-Verlag, 1978.

[27] W. E. Olmstead, S. H. Davis, S. Rosenblat, and W. L. Kath. Bifurcation with memory. *SIAM J. Appl. Math.*, 46(2):171–188, 1986.

[28] D. Roose, K. Lust, A. Champneys, and A. Spence. A Newton-Picard shooting method for computing periodic solutions of large-scale dynamical systems. *Chaos, Solitons and Fractals*, 5(10):1913–1925, 1995.

[29] Y. Saad. *Numerical methods for large eigenvalue problems.* Algorithms and architectures for advanced scientific computing. Manchester University Press, Manchester, 1992.

[30] K. Schmit. Periodic solutions of delay-differential equations. In H.-O.Peitgen and H.-O.Walther, editors, *Functional Differential Equations and Approximation of Fixed Points*, volume 730 of *Lecture Notes in Mathematics*, pages 455–469. Springer-Verlag, 1978.

[31] H. W. Stech. A numerical analysis of the structure of periodic orbits in autonomous functional differential equations. In S.-N. Chow and J. K. Hale, editors, *Dynamics of Infinite Dimensional Systems*, volume F37 of *NATO ASI Series*, pages 325–337. Springer-Verlag, 1987.

[32] H.-O. Walther. A theorem on the amplitudes of periodic solutions of differential delay equations with application to bifurcation. *J. Diff. Eqns.*, 39:369–404, 1978.