15 July 2002

PHYSICS LETTERS A

ELSEVIER

Physics Letters A 299 (2002) 513–521

www.elsevier.com/locate/pla

Parameter estimation in nonlinear delayed feedback systems from noisy data

W. Horbelt *, J. Timmer, H.U. Voss

Freiburger Zentrum für Datenanalyse und Modellbildung, Universität Freiburg, Eckerstr. 1, 79104 Freiburg, Germany

Received 30 January 2002; received in revised form 21 May 2002; accepted 22 May 2002

Communicated by C.R. Doering

Abstract

We propose a method for the estimation of parameters of nonlinear delayed feedback systems from a time series. Being based on the multiple shooting approach it is fairly robust against high levels of observation noise and yields precise parameter estimates. We evaluate its performance using simulated data of the Mackey–Glass equation and present an application to observed time series of an electronic circuit with time delay. © 2002 Elsevier Science B.V. All rights reserved.

PACS: 05.45.-a; 05.45.Tp; 02.30.Ks

Keywords: Parameter estimation; Delay differential equation; Multiple shooting; System identification; Nonlinear delayed feedback system; Mackey–Glass system

1. Introduction

Systems with time-delayed feedback are of practical importance in many different fields ranging from physiology [1,2] to infectious disease models [3,4], cell growth patterns [5] and ring cavity lasers [6–8]. A recent review can be found in [9].

Systems with discrete (as opposed to continuously distributed) delays are in general modelled by delay-differential equations (DDEs). In this Letter we consider deterministic scalar DDEs of the form

$$\dot{x} = f(x, x_\tau), \quad \text{with } x \in \mathbb{R},\ x_\tau(t) := x(t - \tau), \tag{1}$$

where the structure of the right-hand side function f is known but depends on a set of unknown parameters. The aim is to estimate these parameters from measured data.

Various nonparametric methods estimate the time derivative $\dot{x}$ from the data and relate it to x and x_τ. The extremal points of the time series were exploited in [10]. In [11] optimal transformations and the concept of maximal correlation were used. Whilst useful for the estimation of unknown model functions, both methods neglect valuable information about the data, namely that all data points stem from a single continuous trajectory. As a consequence, these methods may be prone to observational noise.

Here we describe a parametric approach, based on maximum likelihood estimation, that can be used for estimation of coefficients in known models. In this

* Corresponding author.
E-mail address: horbelt@physik.uni-freiburg.de (W. Horbelt).

PII: S0375-9601(02)00748-X

method an objective function, quantifying the residuals between the model trajectory and the data, is minimised. In this way observational noise is explicitly included in the model and the entire information about the deterministic nature of the underlying true trajectory is taken into account. However, the cost function usually contains many local minima in the parameter space, apart from the global minimum. Sophisticated strategies have been developed to prevent the optimisation algorithm from stopping in a local minimum [4].

For ordinary differential equations (ODEs) the so-called multiple shooting approach [12] proved to be an important tool to reduce the problem of local minima [13,14]. The pivotal idea in this technique is the formulation of continuity constraints at intermediate points of the fitting interval. An extension of the multiple shooting concept to DDEs requires the notion of continuity to be used in an approximate sense since the state of a DDE includes an infinite number of degrees of freedom.

This Letter proposes such an extension in which the approximation is done via cubic splines. In the following section the formalism of the multiple shooting approach for ODEs is briefly reviewed and the new method for DDEs is described. Its performance is evaluated with simulated data from the Mackey–Glass system in Section 3. Finally the method is applied to measured data from an electronic circuit.

2. Methods

This section provides the methods for the estimation of parameters in DDEs. Before describing the specific DDE extensions, the methods for ODEs are briefly outlined.

2.1. Parameter estimation in ordinary differential equations

Consider a dynamical process described by a nonlinear ODE. To keep notation simple, we restrict ourselves to scalar ODEs, though the method described here was also applied to multivariate data [12–15]. The model trajectory $x(t, x_0, \vec{p})$ depends on the initial value x_0 and the *dynamical parameters* $\vec{p}$ through the initial value problem

$$\dot{x} = f\big(t, x, \vec{p}\big), \quad \text{with } x \in \mathbb{R},\ t \in [t_0, t_N], \tag{2a}$$

$$x(T_0) = x_0. \tag{2b}$$

An experiment gives access to measurements $y_i \in \mathbb{R}$ of the system state at discrete times t_i, i.e.,

$$y_i = x\big(t_i, x_0, \vec{p}\big) + \eta_i, \quad i = 0, \ldots, N. \tag{3}$$

Here, η_i denotes independent normally distributed random numbers with zero mean and variance σ_i^2, accounting for measurement noise.

We aim at estimating the unknown initial value x_0 and the dynamical parameters $\vec{p}$ from the time series $\{y_i\}$. A first approach to accomplish this without the need to estimate derivatives from the data is the *initial value approach*. In this method, initial guesses for x_0 and $\vec{p}$ are chosen. Then, the dynamical equation is solved numerically, and the objective functional $\chi^2(x_0, \vec{p})$ is calculated as the sum of squared residues between the data and the model trajectory, weighted with the inverse variances of the data:

$$\chi^2\big(x_0, \vec{p}\big) = \sum_{i=0}^{N} \frac{1}{\sigma_i^2}\big(y_i - x\big(t_i, x_0, \vec{p}\big)\big)^2. \tag{4}$$

The sought parameters are identified as those minimising $\chi^2(x_0, \vec{p})$. Under the given assumptions they are the maximum likelihood estimates of the true parameters. For the optimisation task, a generalised Gauss–Newton algorithm [12,16] is used which effectively uses a second order approximation to χ^2 even though only first derivatives have to be supplied.

Simulation studies have shown that for many types of dynamics this approach is numerically unstable by yielding a diverging trajectory or stopping in a local minimum [15,17]. This problem can be circumvented by the multiple shooting algorithm. The basic idea of this technique is to consider the task as a multipoint boundary value problem. The fitting interval $[t_0, t_N]$ is partitioned into M subintervals, i.e.,

$$t_0 = T_0 < T_1 < \cdots < T_M = t_N. \tag{5}$$

For each subinterval $[T_j, T_{j+1}]$, *local initial values* $x_j = x(T_j)$ are introduced as additional parameters. The dynamical equation is integrated piecewise and the objective functional $\chi^2(x_0, \ldots, x_{M-1}, \vec{p})$ is evaluated and minimised as in the initial value approach. While the dynamical parameters $\vec{p}$ are unique over the entire interval, the local initial values are optimised separately in each subinterval. Starting guesses for them are suitably chosen to match with the observations. This approach leads to an initially discontinuous trajectory, which is, however, close to the measurements. The final trajectory must of course be continuous, i.e., the computed solution at the end of one subinterval must finally equal the local initial values of the next one:

$$x(T_j - 0) = x(T_j + 0) = x_j, \quad j = 1, \dots, M - 1. \tag{6}$$

These $M - 1$ equations are taken into account as equality constraints in the optimisation procedure. They counterbalance the newly introduced degrees of freedom which thus do not lead to over-fitting. Since only their linearisations are imposed on the update step in the iterative optimisation procedure, the iteration is allowed to proceed to the final continuous solution through "forbidden ground": the iterates will generally be discontinuous trajectories. This freedom allows the method to stay close to the observed data, prevents divergence of the numerical solution and reduces the problem of local minima. More details of the mathematical and implementational aspects of the method are given in [12,17]. Some applications are given in [13,14].

2.2. Extension to delay differential equations

In this subsection the multiple shooting approach is generalised to DDEs. For simplicity only scalar equations with a single time lag τ are considered here. As in the case of ODEs, this method also works for systems of DDEs. The uniqueness of the solution of a DDE with time lag τ requires the specification of an *initial curve* $h_0(t)$ within an interval of length τ. In this way an *initial curve problem* is posed, analogously to Eqs. (2):

$$\dot{x} = f\big(t, x, x_\tau, \vec{p}\big), \quad \text{for } t > T_0, \tag{7a}$$

$$x(t) = h_0(t), \quad \text{for } t \in [T_0 - \tau, T_0]. \tag{7b}$$

We consider DDEs with an arbitrary continuous initial function. As a consequence the state space is infinite-dimensional. If the initial curve is unknown, the question of identifiability arises. Only a finite number of degrees of freedom can be assigned to fit variables.

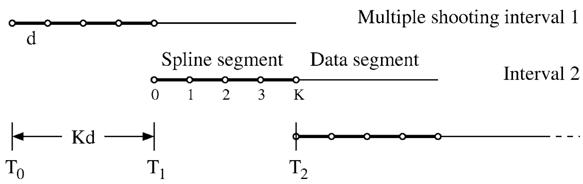

Fig. 1. Schematic diagram of the multiple shooting intervals and their spline and data segments. Bold lines: spline segments. Thin lines: data segments. Open circles: spline knots.

If the multiple shooting technique is used, this problem is multiplied: each subinterval j has its own initial curve h_j of length τ, involving infinitely many degrees of freedom. The purpose of the continuity constraints is to ensure that the final trajectory is a solution of the DDE, i.e., the initial curves of subsequent subintervals must each be consistent with the trajectory on the preceding intervals. Therefore, the subintervals must have an overlap of length τ or more with each other. Let g_j denote the rear segment of the jth subtrajectory that overlaps with the $(j + 1)$st. Continuity would require g_j and h_{j+1} to be exactly equal. This cannot be achieved since h_{j+1} must somehow be represented by a finite number n_s of parameters and g_j will in general not be a member of the corresponding family of functions. In other words, while g_j is an arbitrary vector in a function space, h_{j+1} lies on an n_s-dimensional submanifold of that function space.

Concerning the parameter estimation problem, an additional pitfall for DDEs is described in [18]: a discontinuity of the trajectory x at a time t is echoed as a jump of its first derivative at the later time $t + \tau$. Furthermore, it propagates into the objective function if the delay parameter is to be estimated.

For an efficient albeit practical implementation of the parameter estimation problem we propose the following procedure. The initial curves are parameterised via *cubic splines* [19]. The multiple shooting partitioning Eq. (5) is still used, but the jth subinterval $(j = 1, \dots, M - 1)$ is composed of a *spline segment* $[T_{j-1}, T_j]$ and a data segment $[T_j, T_{j+1}]$. Fig. 1 shows a schematic diagram of this setup. The configuration of the multiple shooting partitioning is chosen such that the spline segments are longer then the maximum expected time lag τ. In each spline segment a mesh of spline knots $\{T_{j-1} + \frac{k}{K}(T_j - T_{j-1}),\ k = 0, \dots, K\}$ is chosen. The spline knots do not necessarily coincide with the data points. The spline function h_j is defined as the piecewise cubic, twice continuously differentiable function that satisfies

$$h_j(T_{j-1} + kd) = s_{jk}, \quad k = 0, \ldots, K, \tag{8}$$

$$\dot{h}_j(T_{j-1} + kd) = \dot{s}_{jk}, \quad k = 0, K, \tag{9}$$

where $\vec{s}_j = (\dot{s}_{j0}, s_{j0}, s_{j1}, \ldots, s_{jK}, \dot{s}_{jK})$ are newly introduced *spline variables*. The number of spline knots $(K + 1)$ is chosen such that the splines are a good approximation of the true trajectory.

The initial curve is the spline function h_j, restricted to the interval $[T_j - \tau, T_j]$, where τ is the actual estimate of the delay parameter. Note that the end of the initial curve is independent of τ. In this way it does not cross any data points when τ is varied and the jump of the trajectory's first derivative at T_j is not reflected in a discontinuity of $\frac{\partial \chi^2}{\partial \tau}$. On the other hand, the trajectory's second derivative jumps at $T_j + \tau$, resulting in a discontinuous $\frac{\partial^2 \chi^2}{\partial \tau^2}$ when $T_j + \tau$ crosses a data point. Quasi-Newton methods or other methods relying on second derivatives of χ^2 could be spoiled by these second order discontinuities, while the Gauss–Newton method that requires only first derivatives is robust against them.

The trajectory is computed in the data segment by integrating Eq. (7b) using the code RETARD [20]. The sensitivities of the trajectory with respect to the parameters, needed in the optimisation process, are obtained by solving the *variational equations* along with Eq. (7b) [17,20]. The cost function is computed by Eq. (4). Then $x(t)$ is projected onto the spline manifold $\{h_{j+1}\}$ of the following subinterval by simply reading out the function values and derivatives, i.e.,

$$r_{jk} = x(T_j + kd), \quad k = 0, \ldots, K, \tag{10a}$$

$$\dot{r}_{jk} = \dot{x}(T_j + kd), \quad k = 0, K. \tag{10b}$$

Now the continuity constraints read

$$\vec{r}_j = (\dot{r}_{j0}, r_{j0}, \ldots, r_{jK}, \dot{r}_{jK}) = \vec{s}_{j+1}, \quad j = 1, \ldots, M-2. \tag{11}$$

As in the case of ODEs, the number of continuity constraints is equal to the number of variables newly introduced in each subinterval. Therefore, K can be chosen large enough to ensure a sufficiently accurate parameterisation of the initial functions, without causing over-fitting problems.

When the starting guesses for the dynamical parameters are far from the true values, a *two-phase procedure* is used. During the first iterations of the optimisation, the spline variables are held fixed because they are expected to be estimated well from the data. After the algorithm has converged for the first time, they are released and fitted together with the other variables until the final convergence is achieved.

This method will be used for estimating parameters in simulated and measured time series in the following two sections. A comprehensive description of the details of the algorithm can be found in [17].

3. Application to simulated data

In this section, the algorithm for estimating parameters in DDEs is applied to the Mackey–Glass equation [1],

$$\dot{x} = \frac{a x_\tau}{1 + x_\tau^c} - bx. \tag{12}$$

3.1. Demonstration of the procedure

A time series of length 1000 was simulated using Eq. (12) with the standard parameters $a = 0.2$, $b = 0.1$ and $c = 10$. The sampling interval was $\Delta t = 1$ and the time lag τ was set to 29.2146, in order to demonstrate that it is not required to be a multiple of the sampling interval. The dynamics was initialised with a constant initial curve $h_0(t) = 0.8$. A transient period of length 10 000 at the beginning was skipped. White Gaussian noise with standard deviation $\sigma = 0.15$ was added to the data, corresponding to a noise level of 50% of the standard deviation of the true trajectory. Then the four parameters (and the spline variables) were estimated using the method described above. The starting guesses for the parameters a, b and c were set to ten-times the true values. The dynamics is much more sensitive to τ than to the other parameters. The intrinsic oscillations of the time series translate into fluctuations of the cost function with respect to τ. Therefore, it is clear that the starting guess for τ should not be too far away from the true value and it was set to 60. The multiple shooting method was applied with 16 subintervals. The spline knots were placed at every third data point. The constants defined above are $d = 3$, $K = 20$ and $M = 16$.

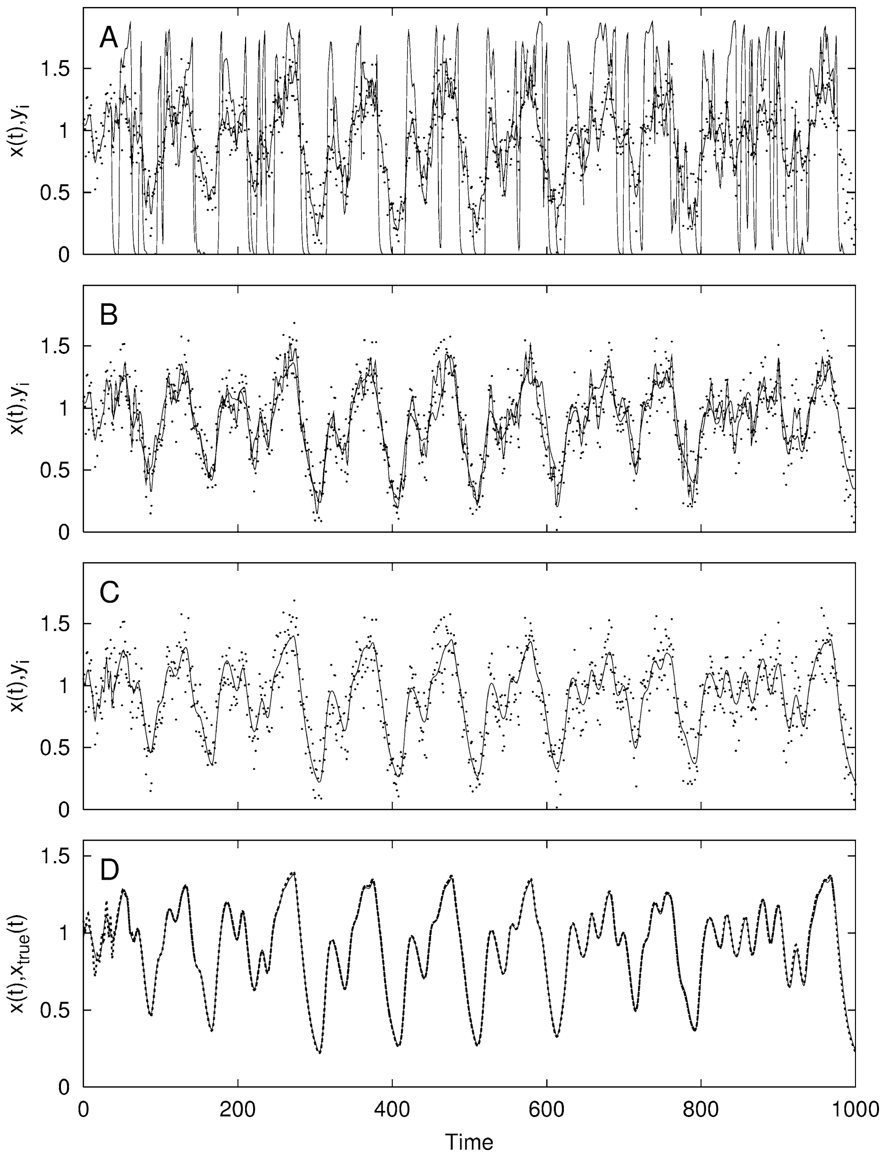

Fig. 2. Estimating the parameters of the Mackey–Glass equation from noisy time series. Points: simulated data. True parameters: $a = 0.2$, $b = 0.1$, $c = 10$, $\tau = 29.2146$, noise level: 50%, sampling interval: $\Delta t = 1$. Lines: model trajectories. At each time two trajectories are overlapping. A: initial situation. Starting guesses of the parameters: $a = 2$, $b = 1$, $c = 100$, $\tau = 60$. B: end of the first phase, in which the spline variables were fixed, after 10 iterations. C: final solution after 6 additional iterations. Estimated parameters: $a = 0.202$, $b = 0.0991$, $c = 9.99$, $\tau = 29.17$. D: true trajectory (thin solid line) and final model trajectory (bold broken line).

Fig. 2A–C show three stages of the iterative optimisation procedure. The initial situation is plotted in Fig. 2A. The trajectories of the individual multiple shooting intervals are overlapping, so that at each interior point, two trajectories can be seen. The spline variables defining the cubic splines were initialised using the data. Then the DDE was integrated over each data segment. In the first phase of the two-phase procedure, the spline variables were held fixed and the parameters were optimised. Panel B shows the situation when the algorithm had converged after 10 iterations. The true trajectory is already imitated well. At this point the estimates for the parameters are $a = 0.154$, $b = 0.0748$, $c = 10.6$ and $\tau = 28.2$. In the second phase the spline variables were freed and fitted together with the parameters. After six additional iterations the algorithm converged finally (panel C). Now the model trajectories are virtually continuous. That means that the spline knots are dense enough to give a reasonable representation of the rear segments of the preceding trajectories. The final value of the objective function is 1065, which is consistent with the number of data points. The final estimates of the parameters are $a = 0.202$, $b = 0.0991$, $c = 9.99$ and $\tau = 29.17$. The fact that τ is a delicate parameter is reflected in its extraordinarily high accuracy: its relative error is only 0.15%.

In Fig. 2D the fit trajectory is compared with the true trajectory. Both are identical to within the line width over the entire interval, except for the beginning. The reason for the small discrepancies in the first oscillation is that some directions in the space of the spline variables represent high-frequency portions that decay rapidly due to the intrinsic low-pass property of differential equations. Thus these spline variables are delicate with respect to noise and their estimates have large confidence intervals. This sign of *over-fitting* would render the result arguable if the construction of the beginning of the model trajectory was a major aim. Here, the principal interest is the accurate estimation of the model parameters, and this aim is achieved by means of the remaining part of the trajectory.

3.2. Systematic test with random starting guesses

In this section we evaluate the performance and the reliability of the method in a simulation study. One thousand time series of length 500 were generated with the parameters $a_0 = 0.2$, $b_0 = 0.1$, $c_0 = 10$, $\tau_0 = 29.2146$ and a noise level of 10%. For the simulation, a constant initial curve $h_0(t) = x_0$ was used and a transient period of length 10 000 was skipped as before, but now x_0 was a random number, uniformly distributed in [0,1]. Thereby not only the noisy time series but also the true trajectory was different in each pass. To each simulated time series the algorithm for estimating the parameters was applied as demonstrated above. The starting guesses for the parameters were drawn randomly from the following probability densities:

$$\rho_a(a) = \frac{1}{a_0}\exp\left(-\frac{a}{a_0}\right), \quad \text{for } a > 0, \tag{13a}$$

$$\rho_b(b) = \frac{1}{b_0}\exp\left(-\frac{b}{b_0}\right), \quad \text{for } b > 0, \tag{13b}$$

$$\rho_c(c) = \frac{1}{c_0 - 1}\exp\left(-\frac{c-1}{c_0-1}\right), \quad \text{for } c > 1, \tag{13c}$$

$$\rho_\tau(\tau) = 0.1, \quad \text{for } 25 \leqslant \tau < 35. \tag{13d}$$

The exponential distributions of a and b allowed large deviations from the true values. The parameter c must be greater than one, otherwise the dynamics would reduce to a fixed point. Therefore, the exponential distribution was shifted to satisfy this constraint. The expectation values for these three distributions are the respective true parameters. The starting guess for τ was uniformly distributed in the interval [25, 35].

The method succeeded in estimating the parameters within 50 iterations in 997 of the 1000 trials. It should be noted that a failed trial does not represent a misleading result, but indicates that the procedure has to be repeated with another portion of data or other starting guesses. Table 1 (middle column) shows the mean estimated parameters and their standard deviation. All parameters were estimated with a relative ac-

Table 1
Mean estimated parameters and standard deviations for two different noise levels, calculated from all respective successful applications of the method. The relative standard deviations are given in parentheses

Parameter	True value	Mean estimate ± standard deviation (relative error)	
		10% noise 997 of 1000 trials successful	50% noise 852 of 1000 trials successful
a	0.2	0.2001 ± 0.002 (1%)	0.2009 ± 0.010 (5%)
b	0.1	0.1001 ± 0.001 (1%)	0.1004 ± 0.0054 (5%)
c	10	10.005 ± 0.07 (0.7%)	10.09 ± 0.52 (5%)
τ	29.2146	29.2146 ± 0.025 (0.09%)	29.22 ± 0.16 (0.5%)

curacy better than 1%, except τ which was determined with a ten-times better precision due to the high sensitivity of the dynamics to changes in τ.

If the time lag had a large uncertainty in a realistic setting, one would apply the method for several different starting guesses. There is a rather large radius of convergence around the true time lag. Therefore, a rather small number of different delays need to be tried, compared with methods that have to scan the entire range of possible values [10,11]. Alternatively one could obtain a good starting guess for τ by using one of the simpler methods described in [10,11,21].

When the simulation study was repeated with the initial value approach rather than multiple shooting, it failed to converge for 945 trials and for only 15 trials the estimated parameters deviated from the true ones by less than 10%. Finally the procedure was repeated again with multiple shooting and with a noise level of 50% (right column of Table 1). In this case the procedure failed for 148 of 1000 realisations. The relative standard errors of the estimated a, b and c are 5% and the standard error of τ is 0.5%. As expected, the errors are five times higher than the respective values for 10% noise.

4. Application to experimental data

In this section the method is applied to a time series from a chaotic electronic oscillator [22]. The experimental setup is drawn in Fig. 3. A transistor and an amplifier form a nonlinear response function f. The low-pass filtered output x is fed into a delay element and coupled back into the input x_τ. The dynamical model reads

$$\dot{x} = -\alpha x + f(x_\tau). \tag{14}$$

The nonlinear response function f is parameterised by the third order polynomial

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3. \tag{15}$$

For the analysis, we use a time series of length 100 ms with a sampling interval of 0.1 ms.

The dynamical parameters are the time lag τ, the damping factor α and the coefficients a_i. The time series shows oscillations with a period of about 4 ms, indicating a time lag in the millisecond range. Therefore, the delay parameter was scanned from 1 to 20 ms

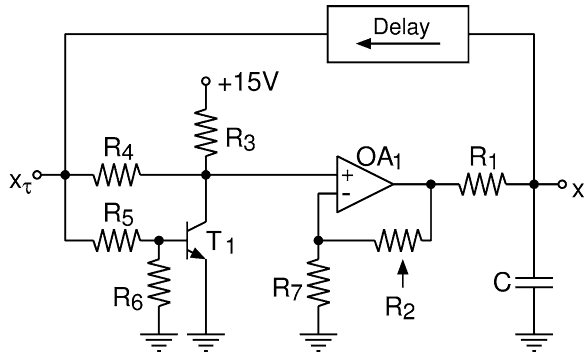

Fig. 3. Schematic diagram of the electronic circuit used in [22], inspired by [23]. The nonlinearity is built up of the transistor T_1, the adjustable amplifier OA_1 and the resistors R_2–R_7. Electronic components: delay line: bucket brigade line MN 3011 with 3328 stages, triggered by MN 3101 (both National Panasonic); OA_1: LM 324N; C = 660 pF; R_1 = 470 kΩ, R_2 = 100 kΩ lin., R_3 = 22 kΩ, R_4 = 4.7 kΩ, R_5 = 10 kΩ, R_6 = 1 kΩ, R_7 = 47 kΩ; T_1 BC 238C.

in steps of 1 ms. Each value was used as a starting guess τ_0 and the estimation procedure was carried out as described earlier. The starting guesses for the other parameters were set to zero. The length of the spline segments was chosen 5 ms longer than τ_0 in order to allow for some variation of τ.

For $\tau_0 = 12$ and 13 ms the method converged within 20 iterations to the values $\tau = 13.3$ ms, $\alpha = 3.7\ \mathrm{ms}^{-1}$, $a_0 = 0.8864\ \mathrm{V\,ms}^{-1}$, $a_1 = -3.982\ \mathrm{ms}^{-1}$, $a_2 = -15.26\ \mathrm{V}^{-1}\,\mathrm{ms}^{-1}$ and $a_3 = -11.54\ \mathrm{V}^{-2}\,\mathrm{ms}^{-1}$. For other starting guesses no convergence was achieved or the final value of the objective function was more than 100 times higher than for the successful fits. Having found good estimates for the parameters, the procedure was started again with the length of the spline segment adjusted to τ and without the multiple shooting technique, in order to ensure that the finite representation of the initial curves does not distort the result. The final estimates are $\tau = 13.2968 \pm 0.00089$ ms, $\alpha = 3.689 \pm 0.015\ \mathrm{ms}^{-1}$, $a_0 = 0.8843 \pm 0.0046\ \mathrm{V\,ms}^{-1}$, $a_1 = -3.973 \pm 0.018\ \mathrm{ms}^{-1}$, $a_2 = -15.209 \pm 0.076\ \mathrm{V}^{-1}\,\mathrm{ms}^{-1}$ and $a_3 = -11.49 \pm 0.13\ \mathrm{V}^{-2}\,\mathrm{ms}^{-1}$. The confidence limits were calculated from the covariance matrix at the solution point. The variance of the data points was estimated from the residuals of the fit. These have a slightly skewed, but otherwise bell-shaped distribution. To test for the validity of the fit, we calculated the autocorrelation function of the residuals, which should decay rapidly,

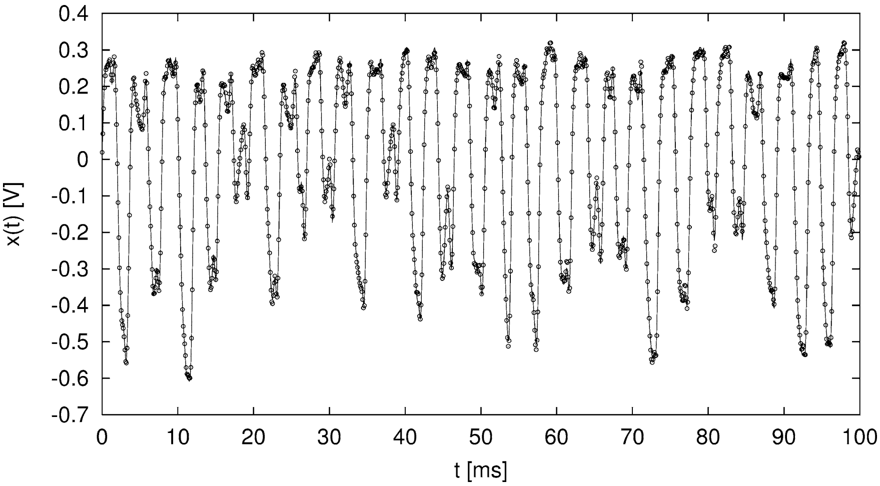

Fig. 4. Comparison of best fit trajectory (solid line) and observed time series (points).

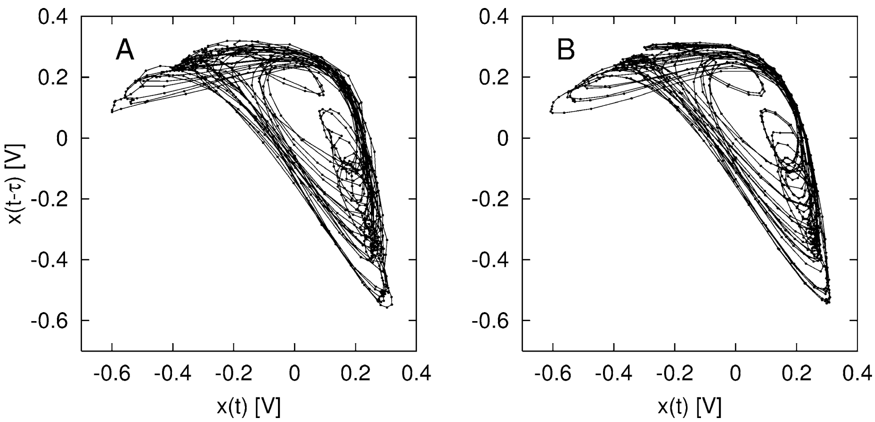

Fig. 5. Phase space reconstruction of the attractor of the system. A: from observed time series. B: from model trajectory.

and indeed, it vanishes already at a time lag of four sampling steps.

The model trajectory is shown together with the time series in Fig. 4. Hardly any difference can be seen in the entire interval. In order to facilitate a comparison, the attractor of the system is reconstructed by a delay embedding, both for the measured time series and for the model trajectory (Fig. 5). The model attractor resembles well the attractor reconstructed from the data, except for some features that are visible in the model, but hidden by noise in the data. Thus apart from yielding the parameters, the modelling procedure performs noise reduction.

5. Conclusion

An advanced method for estimating parameters in delay differential equations is presented. The multiple shooting approach, circumventing the problem of local minima of the cost function, is extended to delay differential equations. The problem of the matching conditions necessary in this context is solved by using cubic splines to parameterise the initial curves and by formulating the continuity of the trajectory in terms of these spline variables.

Using this procedure, parameters of the Mackey–Glass system are estimated accurately for fairly high levels of observation noise and the underlying true trajectory is reconstructed well. A simulation study provides objective performance measures that can be compared with other methods. The time lag can be estimated with an accuracy of 0.5% from only 500 data points with 50% measurement noise. The method has a relatively large radius of convergence and the time lag is not required to be a multiple of the sampling interval.

Finally, the application of the method to measured data from an electronic circuit demonstrates that it is well suited to yield accurate estimates of the parameters of an experimental delayed-feedback system. The obvious next step is to investigate to which extent the robustness of the method and the precision of the estimates are pertained for multivariate systems and for real world systems with large observation error. This is subject of ongoing research.

References

[1] M.C. Mackey, L. Glass, Science 197 (1977) 287.
[2] A. De Gaetano, O. Arino, J. Math. Biol. 40 (2000) 136.
[3] K.L. Cooke, P. van den Driessche, J. Math. Biol. 35 (1996) 240.
[4] G.I. Marchuk, Mathematical Modelling of Immune Response in Infectious Diseases, Vol. 395, Kluwer, Dordrecht, 1997.
[5] C.T.H. Baker, G.A. Bocharov, C.A.H. Paul, F.A. Rihan, J. Math. Biol. 37 (1998) 341.
[6] F.T. Arecchi, G. Giacomelli, A. Lapucci, R. Meucci, Phys. Rev. A 43 (1991) 4997.
[7] R. Hegger, M. Bünner, H. Kantz, A. Giaquinta, Phys. Rev. Lett. 81 (1998) 558.
[8] T. Heil, I. Fischer, W. Elsäßer, J. Opt. B 2 (2000) 413.
[9] G.A. Bocharov, F.A. Rihan, J. Comput. Appl. Math. 125 (2000) 183.
[10] M. Bünner, M. Popp, T. Meyer, A. Kittel, J. Parisi, Phys. Rev. E 54 (1996) R3082.
[11] H. Voss, J. Kurths, Phys. Lett. A 234 (1997) 336.
[12] H.G. Bock, Recent advances in parameter identification techniques for O.D.E., in: P. Deuflhard, E. Hairer (Eds.), Numerical Treatment of Inverse Problems in Differential and Integral Equations, Birkhäuser, Basel, 1983, pp. 95–121.
[13] W. Horbelt, J. Timmer, M. Bünner, R. Meucci, M. Ciofini, Phys. Rev. E 64 (2001) 016222.
[14] J. Timmer, H. Rust, W. Horbelt, H.U. Voss, Phys. Lett. A 274 (2000) 123.
[15] J. Timmer, T. Müller, W. Melzer, Biophys. J. 74 (1998) 1694.
[16] G.A.F. Seber, C.J. Wild, Nonlinear Regression, Wiley, 1989.
[17] W. Horbelt, Maximum likelihood estimation in dynamical systems, Ph.D. Thesis, University of Freiburg, 2001, http://webber.physik.uni-freiburg.de/~horbelt/diss/.
[18] C.T.H. Baker, C.A.H. Paul, SIAM J. Sci. Comput. 18 (1997) 205.
[19] W.H. Press, S.A. Teukolsky, W.T. Vetter, B.P. Flannery, Numerical Recipes in C, 2nd edn., Cambridge University Press, Cambridge, 1992.
[20] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I, Springer, Berlin, 1987.
[21] B.P. Bezruchko, A.S. Karavaev, V.I. Ponomarenko, M.D. Prokhorov, Phys. Rev. E 64 (2001) 056216.
[22] H.U. Voss, Int. J. Bifur. Chaos, in press.
[23] A. Kittel, J. Parisi, K. Pyragas, Physica D 112 (1998) 459.