A Unified Approach to Integration and Optimization of Parametric Ordinary Differential Equations

Daniel Kaschek and Jens Timmer

Abstract Parameter estimation in ordinary differential equations, although applied and refined in various fields of the quantitative sciences, is still confronted with a variety of difficulties. One major challenge is finding the global optimum of a log-likelihood function that has several local optima, e.g. in oscillatory systems. In this publication, we introduce a formulation based on continuation of the log-likelihood function that allows to restate the parameter estimation problem as a boundary value problem. By construction, the ordinary differential equations are solved and the parameters are estimated both in one step. The formulation as a boundary value problem enables an optimal transfer of information given by the measurement time courses to the solution of the estimation problem, thus favoring convergence to the global optimum. This is demonstrated explicitly for the fully as well as the partially observed Lotka-Volterra system.

1 Introduction

Ordinary differential equation (ODE) models play a key role for understanding and predicting the behavior of dynamic systems originating from various disciplines like physics, chemistry or the life sciences. In many cases, these dynamic models depend on parameters that are not known beforehand but need to be determined from measurement data by means of statistical methods. Inference of parameters of dynamic systems from measurement data is commonly realized by optimization of the likelihood function. Optimization is a broad field and many different algorithms have come up over the last decades [1, 7, 11, 13, 14], each of them with problem specific advantages and disadvantages. One characteristic distinction between optimizers is whether they include stochasticity or not. Stochastic optimizers, e.g. evolutionary algorithms [6], particle swarms [9, 12] or simulated annealing [18] are especially valuable for discontinuous likelihood functions where gradient

D. Kaschek (✉) • J. Timmer
Institute of Physics, Freiburg University, Freiburg, Germany
e-mail: daniel.kaschek@physik.uni-freiburg.de; jeti@fdm.uni-freiburg.de

T. Carraro et al. (eds.), *Multiple Shooting and Time Domain Decomposition Methods*, Contributions in Mathematical and Computational Sciences 9,
DOI 10.1007/978-3-319-23321-5_12

information is not available or not defined. On the other hand, many deterministic optimizers employ information about the differentiable structure of the likelihood, i.e. gradient and Hessian information. For differentiable likelihood functions this has the advantage that convergence is achieved much faster. However, this approach has to struggle with other difficulties. If the likelihood function has local optima, the outcome of the optimization procedure depends on the starting point. Once the optimizer is approaching a local optimum, the algorithm will not leave this optimum disregarding the existence of better optima.

The problem of local optima has been addressed by several approaches. It has been shown that a combination of deterministic and stochastic optimization can help escaping local optima and finding the global optimum [15]. Other approaches modify the dynamic system by homotopy transformations [16] introducing a factor λ that allows for a continuous transition between the modified, convex problem and the original problem. Hence another approach is the multiple-shooting method [2]. Most optimizers follow a single-shooting approach, i.e. model trajectories are computed based on given initial values and the outcome is compared to the data. In contrast, the multiple-shooting approach introduces a grid of time-points and initial condition parameters. The optimizer is initialized with discontinuous trajectories and constraints are defined guaranteeing that all trajectories become continuous in the course of optimization.

In our work, we present a reformulation of the optimization problem as a boundary value problem (BVP). The motivation for this approach is twofold. The first argument follows from the history of gradient-based single-shooting optimization for parameter estimation in ordinary differential equations. The performance and accuracy of this method has been enormously increased by solving the ODE together with its' derivatives with respect to the parameters, i.e. the sensitivity equations, in one integration run. This augmentation step allows a fast and accurate computation of the gradient but still evaluation and optimization of the objective function are separate steps. Our aim is to take the next logical step and incorporate even optimization into the ODE integration. The second argument takes up the multiple-shooting idea: the possibility to initialize BVP solvers with prior knowledge like approximate trajectories from measurement data. If the optimization problem is equivalently expressed as a boundary value problem then a good initialization should increase the solver's ability to find the correct solution.

In the following, we show how both objectives can be matched. Our augmentation of the ODE is based on continuation of the log-likelihood function to a differentiable function of time. The resulting system constitutes a BVP. By construction, the solution of this BVP is optimal with respect to the log-likelihood function and it can be obtained by standard numerical BVP solvers. The initialization of the BVP solver allows for an efficient transfer of information provided by the observation data.

2 Methods

We consider a dynamic system defined by ordinary differential equations (ODE),

$$\frac{\mathrm{d}}{\mathrm{d}t}x = f(x,p), \quad x(0) = x_0, \tag{1}$$

with time t, states $x \in \mathbb{R}^n$ and parameters $p \in \mathbb{R}^r$. The extension of the system by $\frac{\mathrm{d}}{\mathrm{d}t}p = 0$ transforms the parameters into usual state variables. For the augmented states $\xi = (x,p)$, parameter estimation becomes an estimation of initial conditions. Furthermore, let $x_{\mathrm{obs}} = (x_1,\ldots,x_m)$, with $m \leq n$, be the observed states and let $\{x_1^D(t_j),\ldots,x_m^D(t_j)\}_j$ denote the time-discrete observation data. The observation data can be approximated by a continuous data function $x_{\mathrm{obs}}^D(t) = (x_1^D(t),\ldots,x_m^D(t))$, e.g. by linear interpolation or spline interpolation. On the other hand, we assume that measurement events for different time points are statistically independent, consequently, the likelihood function

$$L(\xi_0|\{x_{\mathrm{obs}}^D(t_j)\}_j) = \prod_j L_j(\xi_0|x_{\mathrm{obs}}^D(t_j)) \tag{2}$$

factorizes and the negative log-likelihood

$$\ell(\xi_0|\{x_{\mathrm{obs}}^D(t_j)\}_j) = \sum_j -\log L_j(\xi_0, x_{\mathrm{obs}}^D(t_j)) \tag{3}$$

$$\approx \frac{1}{T}\int_0^T r(\xi_0,t)\mathrm{d}t \tag{4}$$

can be approximated by the integral. Here, $r(\xi_0,t)$ denotes the continuous approximation of $-\log L_j(\xi_0, x_{\mathrm{obs}}^D(t_j))$. For standard normally distributed noise, $r(\xi_0,t)$ becomes $\left(x_{\mathrm{obs}}(t) - x_{\mathrm{obs}}^D(t)\right)^2$ which will be used in the following. The argumentation also holds for other noise distributions.

An initial condition vector $\hat{\xi}_0 = (\hat{x}_0,\hat{p}_0) \in \mathbb{R}^{n+r}$ is a local optimum if $\nabla\ell(\hat{\xi}_0, t = T) = 0$ vanishes at the latest observed time point T. Since $\ell(\xi_0, t = 0) = 0$ for all values of ξ_0 at initial time, the gradient $\nabla\ell(\hat{\xi}_0, t = 0) = 0$ vanishes, too. This observation constitutes the boundary condition that needs to be matched for a local optimum, i.e.

$$\nabla\ell(\hat{\xi}_0, 0) = \nabla\ell(\hat{\xi}_0, T) = 0. \tag{5}$$

Each line of Eq. (5) has the potential to determine one parameter value. In order to include this condition into the dynamic system (1), $\nabla\ell$ is derived with respect to time. At this point it is crucial having approximated the negative log-likelihood by

an integral expression:

$$\frac{\mathrm{d}}{\mathrm{d}t}\nabla\ell(\xi_0|x^D_{\mathrm{obs}}(t)) = \frac{1}{T}\frac{\mathrm{d}}{\mathrm{d}t}\int_0^t \nabla r(\xi_0,\tau)\mathrm{d}\tau \tag{6}$$

$$= \frac{1}{T}\nabla r(\xi_0,t) \tag{7}$$

$$= \frac{2}{T}\left(x_{\mathrm{obs}}(t) - x^D_{\mathrm{obs}}(t)\right)^* \mathrm{D}_{\xi_0}x_{\mathrm{obs}}(t). \tag{8}$$

Here, * indicates the transpose and $\mathrm{D}_{\xi_0}x_{\mathrm{obs}}(t)$ denotes the Jacobian of $x_{\mathrm{obs}}(t)$ with respect to the initial conditions ξ_0, also known as the sensitivities of the solution trajectory $x_{\mathrm{obs}}(t)$. The sensitivities are determined by an ODE, too, hence the complete systems reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\xi = f(\xi) \tag{9}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{D}_{\xi_0}\xi = \mathrm{D}_\xi f\,\mathrm{D}_{\xi_0}\xi \tag{10}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\nabla\ell = \frac{2}{T}\left(x_{\mathrm{obs}} - x^D_{\mathrm{obs}}\right)^* \mathrm{D}_{\xi_0}x_{\mathrm{obs}}. \tag{11}$$

The sensitivity equation (10) have fixed initial conditions, $\mathrm{diag}(\mathbb{1}_{n+r})$, with the identity matrix $\mathbb{1}_{n+r} \in \mathbb{R}^{(n+r)\times(n+r)}$. The gradient equations (11) have both zero initial *and* final condition, see Eq. (5), a boundary constraint that fully determines the initial values of the augmented states in Eq. (9). On the other hand, these are the parameters and initial conditions we seek to estimate. Hence, the desired values $\hat{\xi}_0$ optimizing the negative log-likelihood are part of the solution of the two-point boundary value problem, Eqs. (9)–(11). Compared to gradient based single-shooting methods, Eq. (11) represents the pivotal difference. It translates optimization into the ambit of integration. This is the principle behind our optimization approach.

The solution of the two-point boundary value problem is obtained by the Fortran 77 code TWPBVP [4, 5], available from the Netlib repository. The method used in TWPBVP is a deferred correction method based on mono-implicit Runge-Kutta formulas and adaptive mesh refinement. The deferred correction algorithm uses the trapezoidal rule to obtain a first approximation to the required solution. Finite difference approximations to the local truncation error are then added onto this low order solution, increasing the accuracy of the solution repeatedly [3].

3 Example

In the following, we examine the Lotka-Volterra equations [8, 10, 17]. They give a basic description of the predator and prey population dynamics. The system is

defined by two differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}A = A(\alpha - \beta B), \tag{12}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}B = -B(\gamma - \delta A), \tag{13}$$

where A and B correspond to prey and predator respectively. The parameters α and β describe prey reproduction and reduction, γ and δ describe predator extinction and reproduction. For non-zero initial condition, the solution of Eqs. (12)–(13) is a sustained oscillation.

In the first part of the study, it is assumed that both populations are observed. Observation data is simulated by numerically integrating the ODE system with $A_0 = 1$, $B_0 = 1.2$, $\alpha = 0.45$, $\beta = 0.5$, $\gamma = 0.3$, $\delta = 0.1$ and adding Gaussian noise with $\sigma = 0.15$ to the solution. Subsequently, the parameter values are sought to be recovered from randomly chosen initial parameter guesses. The initialization of the BVP solver, i.e. the grid of initially assumed values for each of the variables in Eqs. (9)–(11), is obtained by integrating the sensitivity equation (10) with the initial parameter guess and the data interpolations as input trajectories. The gradient is assumed to vanish over the entire range.

Independently of the initial parameter values, the BVP solver converges to the same solution. The result for one representative data set is shown in Fig. 1. The *States* panel shows the solutions of the state variables A and B together with the data points and error bars. As expected, the predator and prey trajectories hit about 67 % of the error bars. In the *Parameters* panel, the solutions of the state variables α, β, γ and δ are shown on a logarithmic scale, i.e. the dynamic parameters. The dots indicate the values that have been used for simulation. The *Sensitivities* panel shows the sensitivity trajectories which are typical for oscillating systems, i.e. oscillations with increasing amplitude. Finally, in the *Negative log-likelihood gradient* panel, the gradient solution is plotted. The time scale of gradient changes is determined by the sampling density of the simulated time course. It hits the ground line in the end point as desired, guaranteeing an optimum.

The BVP method has been tested systematically against a single-shooting approach based on the Levenberg-Marquardt algorithm, implemented in the MINPACK Fortran 77 package. For different simulated data sets, both, BVP method and single-shooting method have been applied to the same random set of initial parameter guesses. In order to avoid that, by chance, parameter vectors are too similar, we employed Latin hypercube sampling with a hypercube covering 4 orders of magnitude around the true parameter values. Both optimization approaches failed convergence a number of times in which case 10^6 was assigned as value to the negative log-likelihood. Figure 2 shows the first 200 sorted negative log-likelihood values of the total 300 initial guesses for both approaches. The single-shooting approach gets stuck in different local optima and finds the global optimum only in 4 % of the cases. In contrast, the BVP method proves to be robust against different initial guesses for the parameter values. The solution converges either to the global

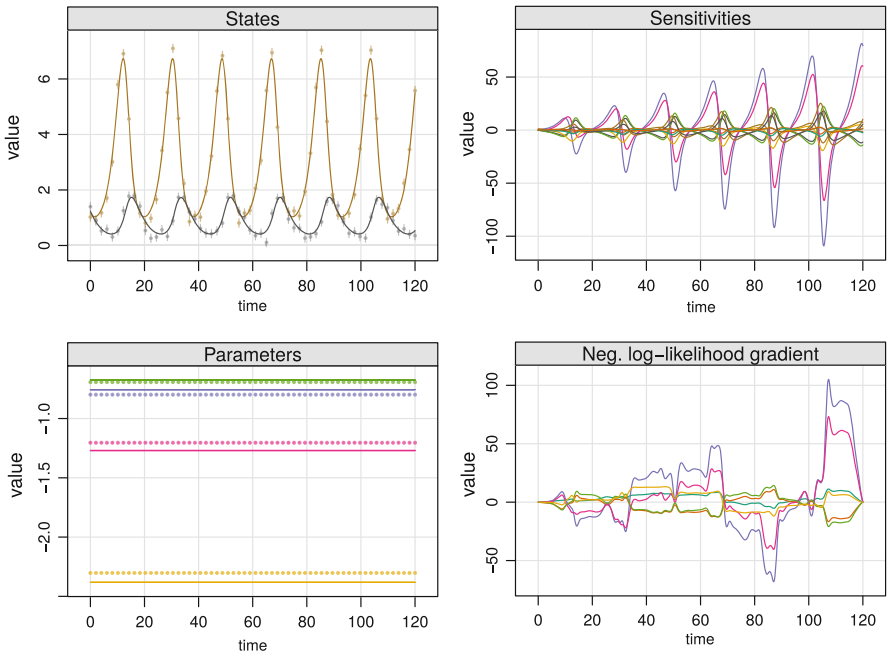

Fig. 1 Solution generated by the BVP solver. The four panels show state solutions with simulated data points, parameter solutions on a logarithmic scale with true values as *dots*, sensitivity solutions and the gradient of the negative log-likelihood

minimum or it fails convergence. It is 6 times more efficient in finding the best optimum. Figure 2 also gives some indication about parameter convergence regions. For the BVP method, the set of initial parameters that finally converged to the best parameter value covers almost the total range. However, fewer initial guesses with α and δ larger than 1 lead to a successful reconstruction of the BVP solution. The broad plateau of local optima for the single-shooting method is reflected in a clear shift of final parameter values and a certain number of randomly distributed final parameters.

In a second step, the observation of B is omitted and β is fixed to 1 in order to keep the system identifiable. Analogously to the fully observed system, data sets have been simulated and Gaussian noise has been added. The comparison between the single-shooting method and the BVP method is shown in Fig. 3a. The plots indicate that the situation becomes more intricate if only one state is observed. The convergence rate drops below 2 % for both approaches and the BVP method reconstructs a variety of local optima, each of them with an almost identical negative log-likelihood value. From the scatter plots in Fig. 3, two conclusions can be drawn for the BVP method: First, whenever we found the global minimum, the parameters were initially situated in the negative orthant, and second, the final parameters

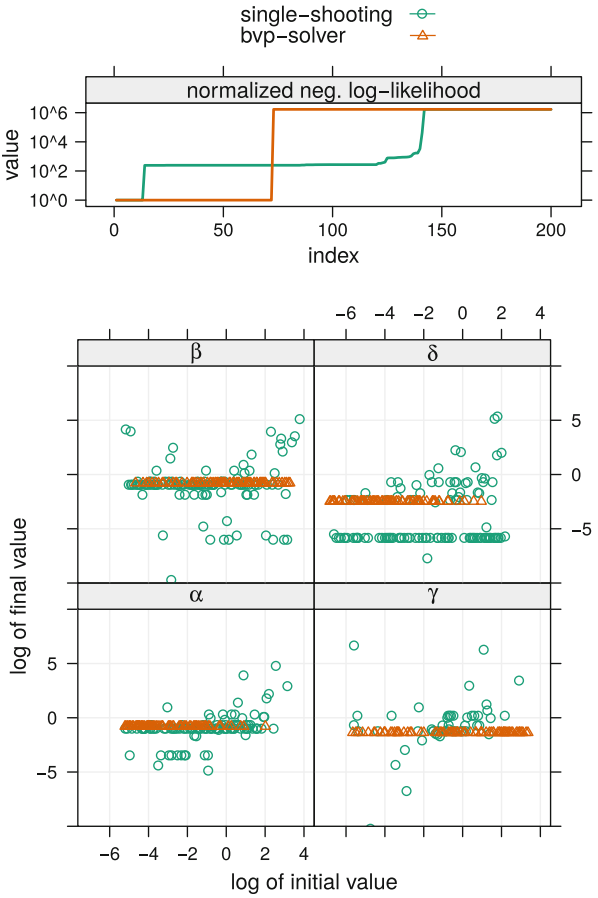

Fig. 2 Comparison of single-shooting and BVP method tested on the fully observed Lotka-Volterra system. Each method has been applied to simulated data sets and for each data set, initial parameter vectors have been generated by Latin hypercube sampling covering a range of 4 orders of magnitude around the true parameter values. The resulting negative log-likelihood values were sorted, normalized by the smallest value and plotted on a logarithmic scale. In the scatter plots, initial parameter vectors are plotted against final parameter values for each optimization that resulted in a negative log-likelihood value smaller than 10^3

corresponding to the local optima form a submanifold with boundary in parameter space.

Figure 3b shows the same picture for initial guesses starting from the negative orthant only. In agreement with the expectation, the number of BVP solutions corresponding to the global optimum increases considerably and exceeds the success rate of the single-shooting method by a factor of 5.

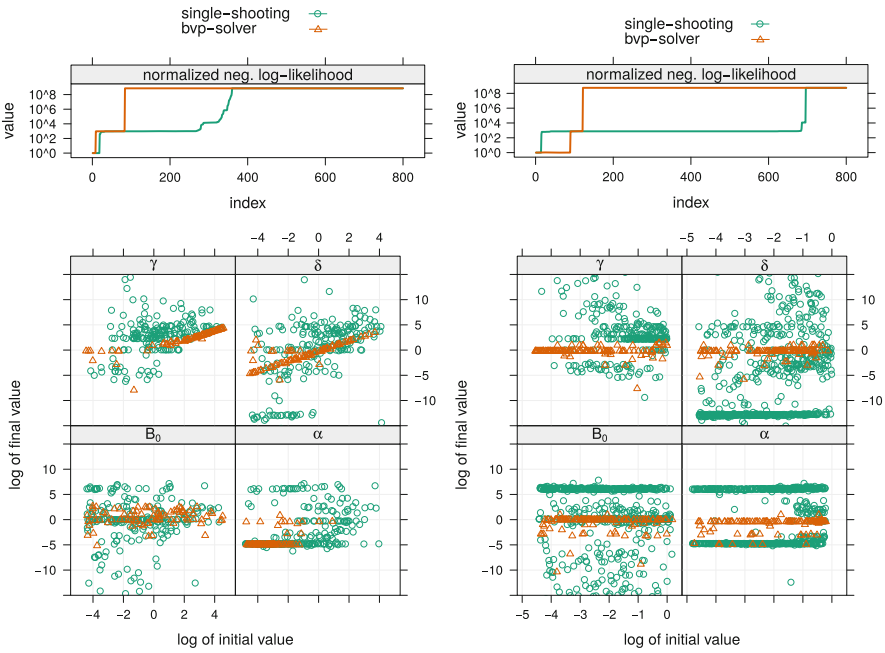

Fig. 3 Comparison of single-shooting and BVP method tested on the partially observed Lotka-Volterra system. Each method has been applied to simulated data sets and for each data set, initial parameter vectors have been generated by Latin hypercube sampling. Column (**a**) shows the results for initial parameter vectors covering a range of 4 orders of magnitude around the true parameter values. For column (**b**), initial parameters were restricted to values smaller than 10^0. In both cases, the resulting negative log-likelihood values were sorted, normalized by the smallest value and plotted on a logarithmic scale. In the scatter plots, initial parameter vectors are plotted against final parameter values for each optimization that resulted in a negative log-likelihood value smaller than 10^3

4 Conclusion

In case of a partially observed system, the success rate of the BVP method depends on favorable initial conditions. Whereas the single-shooting algorithm performed equally badly over the entire parameter space, for the boundary value approach, it was possible to identify an attractive basin resulting in a considerably increased convergence rate.

From the fully observed Lotka-Volterra system, we conclude that the boundary value approach is excellently suited as optimization approach if numerous observables are available. In this case, it clearly outperforms the single-shooting Levenberg-Marquardt algorithm in terms of convergence to the global optimum. The strength of the presented optimization approach is its ability to exploit the measured time courses in a natural way. This favors convergence to the global optimum. Unlike single-shooting approaches, convergence to local optima or

false convergence claims are efficiently reduced. On the other hand, the deferred correction algorithm seemed to be very sensitive to the grid initialization by initial parameter and state guesses, manifesting in a large number of non-convergent attempts. This problem increased with the size of the time domain. At this point, a multiple shooting algorithm, being based on time-domain decomposition, is expected to be more stable and to provide a higher convergence rate.

In summary, we presented a reformulation of the estimation problem as a boundary value problem which, in turn, is enabled by continuation of the negative log-likelihood function to a time-differentiable function. This restatement elegantly incorporates optimization and ODE solution in one task. By nature of the boundary value problem, an initial guess for all state variables needs to be presented to the numerical solver. The initialization by measured time-courses carries exactly the information that is necessary to make the algorithm converge to the best optimum.

Acknowledgements This work was supported by the MIP-DILI project, Innovative Medicines Initiative Joint Undertaking under grant agreement No. 115336. We thank our colleague Marcus Rosenblatt for supporting the computational implementation.

References

1. Amritkar, R.E.: Estimating parameters of a nonlinear dynamical system. Phys. Rev. E **80**(4), 047,202 (2009). doi:10.1103/PhysRevE.80.047202
2. Bock, H.G., Kostina, E., Schlöder, J.P.: Numerical methods for parameter estimation in nonlinear differential algebraic equations. GAMM-Mitt. **30**(2), 376–408 (2007)
3. Cash, J.: A variable order deferred correction algorithm for the numerical solution of nonlinear two point boundary value problems. Comput. Math. Appl. **9**(2), 257–265 (1983)
4. Cash, J., Mazzia, F.: A new mesh selection algorithm, based on conditioning, for two-point boundary value codes. J. Comput. Appl. Math. **184**(2), 362–381 (2005)
5. Cash, J., Wright, M.H.: A deferred correction method for nonlinear two-point boundary value problems: implementation and numerical evaluation. SIAM J. Sci. Stat. Comput. **12**(4), 971–989 (1991)
6. Goswami, G., Liu, J.S.: On learning strategies for evolutionary monte carlo. Stat. Comput. **17**(1), 23–38 (2007). doi:10.1007/s11222-006-9002-y
7. Horbelt, W., Timmer, J., J. Bünner, M., Meucci, R., Ciofini, M.: Identifying physical properties of a CO_2 laser by dynamical modeling of measured time series. Phys. Rev. E **64**(1), 016,222 (2001). doi:10.1103/PhysRevE.64.016222
8. Lotka, A.J.: Contribution to the theory of periodic reactions. J. Phys. Chem. **14**(3), 271–274 (1909). doi:10.1021/j150111a004
9. Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: simpler, maybe better. IEEE Trans. Evol. Comput. **8**(3), 204–210 (2004). doi:10.1109/TEVC.2004.826074
10. Parker, M., Kamenev, A.: Extinction in the lotka-volterra model. Phys. Rev. E **80**(2), 021,129 (2009). doi:10.1103/PhysRevE.80.021129
11. Parlitz, U.: Estimating model parameters from time series by autosynchronization. Phys. Rev. Lett. **76**(8), 1232–1235 (1996). doi:10.1103/PhysRevLett.76.1232
12. Peng, H., Li, L., Yang, Y., Liu, F.: Parameter estimation of dynamical systems via a chaotic ant swarm. Phys. Rev. E **81**(1), 016,207 (2010). doi:10.1103/PhysRevE.81.016207
13. Sitz, A., Schwarz, U., Kurths, J., Voss, H.U.: Estimation of parameters and unobserved components for nonlinear systems from noisy time series. Phys. Rev. E **66**(1), 016,210 (2002). doi:10.1103/PhysRevE.66.016210

14. Sohl-Dickstein, J., Battaglino, P.B., DeWeese, M.R.: New method for parameter estimation in probabilistic models: Minimum probability flow. Phys. Rev. Lett. **107**(22), 220,601 (2011). doi:10.1103/PhysRevLett.107.220601
15. Villaverde, A.F., Egea, J.A., Banga, J.R.: A cooperative strategy for parameter estimation in large scale systems biology models. BMC Syst. Biol. **6**(1), 75 (2012). doi:10.1186/1752-0509-6-75
16. Vyasarayani, C., Uchida, T., McPhee, J.: Single-shooting homotopy method for parameter identification in dynamical systems. Phys. Rev. E **85**(3) (2012). doi:10.1103/PhysRevE.85.036201
17. Wang, M.X., Lai, P.Y.: Population dynamics and wave propagation in a lotka-volterra system with spatial diffusion. Phys. Rev. E **86**(5), 051,908 (2012). doi:10.1103/PhysRevE.86.051908
18. Xiang, Y., Gong, X.G.: Efficiency of generalized simulated annealing. Phys. Rev. E **62**(3), 4473–4476 (2000). doi:10.1103/PhysRevE.62.4473