Noname manuscript No.
(will be inserted by the editor)

Numerical Methods for Computing Sensitivities for ODEs and DDEs

Jonathan Calver · Wayne Enright

Received: date / Accepted: date

Abstract We investigate the performance of the adjoint approach and the variational approach for computing the sensitivities of the least squares objective function commonly used when fitting models to observations. We note that the discrete nature of the objective function makes the cost of the adjoint approach for computing the sensitivities dependent on the number of observations. In the case of ODEs, this dependence is due to having to interrupt the computation at each observation point during numerical solution of the adjoint equations. Each observation introduces a jump discontinuity in the solution of the adjoint differential equations. These discontinuities are propagated in the case of DDEs, making the performance of the adjoint approach even more sensitive to the number of observations for DDEs. We quantify this cost and suggest ways to make the adjoint approach scale better with the number of observations. In numerical experiments, we compare the adjoint approach with the variational approach for computing the sensitivities.

Keywords Ordinary differential equations · Delay differential equations · Adjoint method · Variational equations · Sensitivities

Introduction

Ordinary differential equations (ODEs) have long been used in the physical sciences to help us better understand the world around us. More recently, delay differential equations (DDEs) have been successfully applied to model phenomena in a variety of applications, such as population dynamics [7,5],

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

J. Calver · W. Enright
Department of Computer Science, University of Toronto, Toronto, ON, M5S 3G4, Canada
E-mail: {calver,enright}@cs.toronto.edu

epidemics [6], and chemical systems with feedback [11]. Central to the modeling process is the computation of sensitivities, which provide key information for a variety of tasks related to quantifying uncertainty, determining optimal parameters to fit observed data, and designing experiments to collect appropriate data to facilitate model fitting. In any case, it is desirable to be able to compute these sensitivities efficiently and with sufficient accuracy for the application.

We consider two cases where the need for sensitivities arise. In the problem of determining the best fit values of model parameters, we are interested in computing the sensitivity of an objective function to changes in the model parameters. If these sensitivities are computed sufficiently accurately, they can be used in a gradient based optimization scheme to efficiently find a local minimum of the objective function.

The second use for sensitivities is to quantify how sensitive the model is to changes in its parameters when the optimal parameters are known. While similar to the previous case of parameter estimation, this is different in that we need to compute how sensitive the model, not the objective function, is to changes in the parameter values. Note that the model sensitivities are a matrix valued quantity, while the sensitivities for a scalar objective function are a vector valued quantity.

In section one, we introduce the general forms of the models we consider and the least squares (LSQ) objective function. The variational and adjoint approaches are then described in section two. The third section describes our numerical experiments and suggests ways to deal with limitations of the adjoint approach that arise in the case of LSQ. The numerical experiments use DDEM - a C++ framework for solving and analyzing DDEs [14]. In the fourth section, we discuss how the cost of each approach scales with the number of delays in the model and we suggest ways to exploit parallelism. The final section summarizes our results and outlines future work.

Related Work

In [10], the authors look at comparing the adjoint, variational, and finite difference methods for computing objective function sensitivities for IVPs, focusing on cases where the adjoint method is most suitable. The adjoint method is best suited for cases where the objective function is scalar and the model consists of more parameters than state variables. In this work, our focus is on how the adjoint and variational methods scale with the number of observations in the LSQ objective function.

In [8], Bock considers the benefits of applying the principle of internal differentiation (as opposed to external differentiation) to computing the sensitivities of DDEs. They look at implementing the forward approach by differentiating an implicit CRK formula directly, as opposed to setting up the system of DDEs for the solver to integrate, and taking advantage of the structure of the resulting equations. Issues of error control are discussed and their approach benefits from being able to conveniently control the error of the sensitivities being computed when low order approximations are used.

Explicit CRK solvers are also used in the fortran library, DENSERKS [1], to efficiently interpolate the forward solution during simulation of the associated adjoint IVP.

For a good reference on the variational and adjoint methods for DAEs and PDEs, we refer the reader to [9].

1 Mathematical Setting

1.1 ODE definition

We consider the parameterized initial value problem (IVP),

$$\begin{aligned} \dot{\boldsymbol{y}}(t) &= \boldsymbol{f}(t, \boldsymbol{y}(t), \boldsymbol{p}), \\ \boldsymbol{y}(0) &= \boldsymbol{y}_0, \\ t &\in (0, T), \end{aligned} \tag{1}$$

where $\boldsymbol{y}$ is the state vector of dimension n_y, $\boldsymbol{p}$ is the constant vector of model parameters of dimension n_p, and $\boldsymbol{y}_0$ are the initial conditions of the state vector - each component of which may be a parameter appearing in $\boldsymbol{p}$. We will denote the solution of (1) for a specific $\boldsymbol{p}$ by $\boldsymbol{y}(t, \boldsymbol{p})$. For convenience, when it is clear, we will use $\boldsymbol{y}(t)$ to represent $\boldsymbol{y}(t, \boldsymbol{p})$.

1.2 DDE definition

We restrict ourselves to systems of DDEs with constant lags, in which case we are considering the IVP,

$$\begin{aligned} \dot{\boldsymbol{y}}(t) &= \boldsymbol{f}(t, \boldsymbol{y}(t), \boldsymbol{y}(t-\tau_1), \ldots, \boldsymbol{y}(t-\tau_{n_d}), \boldsymbol{p}), \\ t &\in (0, T), \\ \boldsymbol{y}(t) &= \boldsymbol{h}(t, \boldsymbol{p})\ ;\ t < 0, \end{aligned} \tag{2}$$

where each τ_r, $r = 1, \ldots, n_d$, is a constant delay that may appear in our parameter vector, $\boldsymbol{p}$, and $\boldsymbol{h}(t, \boldsymbol{p})$ is the history function. Note that, as with (1), $\boldsymbol{y}$ and $\boldsymbol{h}$ are vector valued functions of dimension n_y.

1.3 Data

We assume that a series of measurements (or observations) of the state vector is given,

$$\tilde{\boldsymbol{y}}_j(t_i) = \boldsymbol{y}_j(t_i, \boldsymbol{p}^*) + \text{noise, for } \ i = 1, \ldots, n_o;\ j = 1, \ldots, n_y,$$

where n_o is the number of observation points and $\boldsymbol{y}_j(t_i, \boldsymbol{p}^*)$ denotes the true solution of the IVP at time t_i corresponding to the optimal value, $\boldsymbol{p}^*$, that best-fits the observed data. For least squares, the noise is assumed to follow a normal distribution.

1.4 Model Sensitivities

Model sensitivities refer to how sensitive the solution of the model is to its parameters. In our case, for a given time, $t \in [0, T]$, this is the matrix valued quantity,

$$\frac{\partial \boldsymbol{y}(t, \boldsymbol{p})}{\partial \boldsymbol{p}} \equiv \boldsymbol{y}_{\boldsymbol{p}}(t, \boldsymbol{p}), \tag{3}$$

For convenience, we will use $\boldsymbol{y}_{\boldsymbol{p}}(t)$ to represent $\boldsymbol{y}_{\boldsymbol{p}}(t, \boldsymbol{p})$.

1.5 Least Squares Parameter Estimation (Inverse Problem)

As our main application, we consider least squares (LSQ) parameter estimation. For a given value of p, the standard LSQ objective function is,

$$S(\boldsymbol{p}) = \sum_{i=1}^{n_o} \frac{||\tilde{\boldsymbol{y}}(t_i) - \boldsymbol{y}(t_i, \boldsymbol{p})||^2}{2}, \tag{4}$$

where the Euclidean norm is used.

2 Methods

2.1 Variational Approach

The first approach we will describe is the variational approach, sometimes referred to as the forward approach. This standard approach for computing model sensitivities directly approximates the variational equations, which are characterized or defined by an ODE satisfied by $\boldsymbol{y}_{\boldsymbol{p}}(t)$.

These equations are easily derived by taking the time derivative of $\boldsymbol{y}_{\boldsymbol{p}}(t)$,

$$\begin{aligned}
\frac{d}{dt}\boldsymbol{y}_{\boldsymbol{p}}(t) &= \frac{\partial}{\partial \boldsymbol{p}} \frac{d\boldsymbol{y}}{dt}(t) \\
&= \frac{\partial}{\partial \boldsymbol{p}} \boldsymbol{f}(t, \boldsymbol{y}(t, \boldsymbol{p}), \boldsymbol{p}) \\
&= \boldsymbol{f}_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)
\end{aligned}$$

This matrix valued ODE can be solved simultaneously with the original system, (1), with the initial conditions, $\boldsymbol{y_p}(0)$, whose (i, j) entry is,

$$\frac{\partial \boldsymbol{y}_i}{\partial \boldsymbol{p}_j}(0) = \begin{cases} 1, & \text{if } \boldsymbol{p}_j \text{ is the initial condition for } \boldsymbol{y}_i \\ 0, & \text{otherwise} \end{cases}.$$

Approximating the solution of this system directly gives us the model sensitivities for any t. The sensitivity of the LSQ objective function with respect to the parameters can then be approximated from (1) and (4) as,

$$\frac{\partial S(\boldsymbol{p})}{\partial \boldsymbol{p}} = \sum_{i=1}^{n_o} (\tilde{\boldsymbol{y}}(t_i) - \boldsymbol{y}(t_i, \boldsymbol{p}))^T \boldsymbol{y_p}(t_i, \boldsymbol{p}). \tag{5}$$

2.1.1 Variational Approach for DDEs

For constant lag DDEs, the variational approach results in the following neutral DDE, which we obtain by again taking the time derivative of $\boldsymbol{y_p}(t)$,

$$\begin{aligned} \frac{d}{dt}\boldsymbol{y_p}(t) &= \frac{\partial}{\partial \boldsymbol{p}} \frac{d\boldsymbol{y}}{dt}(t) \\ &= \frac{\partial}{\partial \boldsymbol{p}} \boldsymbol{f}(t, \boldsymbol{y}(t, \boldsymbol{p}), \boldsymbol{y}(t-\tau, \boldsymbol{p}), \boldsymbol{p}) \\ &= \boldsymbol{f_y}(t)\boldsymbol{y_p}(t) + \boldsymbol{f_p}(t) + \boldsymbol{f_\nu}(t)\boldsymbol{y_p}(t-\tau) - \boldsymbol{y}'(t-\tau)\tau_{\boldsymbol{p}}, \end{aligned}$$

where $\boldsymbol{\nu} = \boldsymbol{y}(t-\tau)$. This matrix valued DDE can be approximated simultaneously with the original system,(2), with the initial conditions, $\boldsymbol{y_p}(0)$, whose (i, j) entry is,

$$\frac{\partial \boldsymbol{y}_i}{\partial \boldsymbol{p}_j}(0) = \begin{cases} 1, & \text{if } \boldsymbol{p}_j \text{ is the initial condition for } \boldsymbol{y}_i \\ 0, & \text{otherwise} \end{cases}.$$

The corresponding history function is given by,

$$\frac{\partial \boldsymbol{y}_i}{\partial \boldsymbol{p}_j}(t) = \frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{p}_j}(t), \text{for } t < 0.$$

A limitation of this approach is that the variational system consists of $n_y + n_y n_p$ differential equations. A significant advantage of this approach compared to the adjoint approach is that it extends in a straightforward way to general systems of DDEs and not just to the special class we are considering here. For details, see [16].

2.2 Adjoint Approach

For a more complete discussion of the adjoint method, we refer the reader to [4]. We first present a derivation of the adjoint method for systems of IVPs. In the following, we consider objective functions of the form,

$$G(\boldsymbol{y},\boldsymbol{p}) = G(\boldsymbol{y}(\boldsymbol{p})) = \int_0^T g(\boldsymbol{y}(t,\boldsymbol{p}))\,dt, \tag{6}$$

where we are restricting ourselves to objective functions that only depend on $\boldsymbol{p}$ through $\boldsymbol{y}(s,\boldsymbol{p})$. Let $\boldsymbol{\lambda}^T(t)$ be any vector valued function of dimension n_y, defined for $t \in [0,T]$. Now, consider the perturbed objective function,

$$J(\boldsymbol{p}) = G(\boldsymbol{y}(\boldsymbol{p})) + \int_0^T \boldsymbol{\lambda}^T(t)\big(\dot{\boldsymbol{y}}(t,\boldsymbol{p}) - \boldsymbol{f}(t,\boldsymbol{y}(t,\boldsymbol{p}),\boldsymbol{p})\big)\,dt. \tag{7}$$

Note that the term we have added is zero, since $\boldsymbol{y}(t)$ satisfies the ODE, (1). Taking the derivative with respect to the parameters, we obtain,

$$\begin{aligned} J_{\boldsymbol{p}} &= \frac{dG}{d\boldsymbol{p}} + \int_0^T \boldsymbol{\lambda}^T(t)\left(\frac{d\dot{\boldsymbol{y}}}{d\boldsymbol{p}}(t) - \frac{\partial}{\partial \boldsymbol{p}}\Big[\boldsymbol{f}(t,\boldsymbol{y}(t,\boldsymbol{p}),\boldsymbol{p})\Big]\right)dt \\ &= \frac{dG}{d\boldsymbol{p}} + \int_0^T \boldsymbol{\lambda}^T(t)\Big(\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt. \end{aligned} \tag{8}$$

From (6),

$$\begin{aligned} \frac{dG}{d\boldsymbol{p}} &= \frac{\partial}{\partial \boldsymbol{p}}\Big[\int_0^T g(\boldsymbol{y}(t,\boldsymbol{p}))\,dt\Big] \\ &= \int_0^T g_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt, \end{aligned}$$

and therefore, from (8),

$$J_{\boldsymbol{p}} = \int_0^T \left(g_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) + \boldsymbol{\lambda}^T(t)\Big(\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\right)dt. \tag{9}$$

Using integration by parts, we can express the integral term involving $\boldsymbol{\lambda}^T(t)\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t)$ as,

$$\int_0^T \boldsymbol{\lambda}^T(t)\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t)\,dt = \Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T - \int_0^T \dot{\boldsymbol{\lambda}}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt. \tag{10}$$

From (10) and (9), we obtain, after re-arranging terms,

$$J_{\boldsymbol{p}} = \int_0^T \Big(g_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t) - \dot{\boldsymbol{\lambda}}^T(t)\Big)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt - \int_0^T \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{p}}(t)\,dt + \Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T. \tag{11}$$

The adjoint system is defined by requiring that $\boldsymbol{\lambda}^T(t)$ be the solution of the IVP:

$$\dot{\boldsymbol{\lambda}}^T(t) = g_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t)\,; \boldsymbol{\lambda}^T(T) = \mathbf{0}. \tag{12}$$

This choice for $\boldsymbol{\lambda}^T(t)$ eliminates the integral involving $\boldsymbol{y}_{\boldsymbol{p}}(t)$ in (11) and results in the sensitivities being given by,

$$\begin{aligned} J_{\boldsymbol{p}} &= -\int_0^T \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{p}}(t)\,dt + \Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T \\ &= -\int_0^T \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{p}}(t)\,dt - \boldsymbol{\lambda}^T(0)\boldsymbol{y}_{\boldsymbol{p}}(0). \end{aligned} \tag{13}$$

In some applications, we might not be interested in the sensitivity of G, but rather in the sensitivity of g at time T. In this case, if we take derivatives of (12) and (13) with respect to T, we obtain,

$$\dot{\boldsymbol{\lambda}}_T^T(t) = -\boldsymbol{\lambda}_T^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t)\,; \boldsymbol{\lambda}_T^T(T) = g_{\boldsymbol{y}}(T), \tag{14a}$$

$$\frac{dg}{d\boldsymbol{p}} = -\int_0^T \boldsymbol{\lambda}_T^T(t)\boldsymbol{f}_{\boldsymbol{p}}(t)\,dt + \boldsymbol{\lambda}_T^T(0)\frac{d\boldsymbol{y}}{d\boldsymbol{p}}(0), \tag{14b}$$

Note that the above equations are defined with t varying from T to 0. Defining $x = T - t$, we can return to the standard situation where the independent variable varies from 0 to T.

For our purposes, there are two objective functions of interest. The first is the LSQ objective function, the second corresponds to model sensitivities. In the case of LSQ, we can either use (12-13) or (14) above. For model sensitivities, we are trying to obtain the sensitivities of the model to the parameters at a specific t, so we use (14). We will now investigate in more detail these two forms.

2.2.1 Case of Model Sensitivities

In this case, with $g(s, \boldsymbol{y}(s, \boldsymbol{p})) = \boldsymbol{y}_j(t_i, \boldsymbol{p})$, we have $g_{\boldsymbol{y}} = \boldsymbol{e}_j$, where $\boldsymbol{e}_j$ is one for the j^{th} component and zero for the other components. This will give us the sensitivity of the $\boldsymbol{y}_j$ at time t_i. While this is fine mathematically, it turns out this is inefficient in practice. The difficulty is that since $\boldsymbol{y}$ has n_y components, we must apply the adjoint approach to each component of $\boldsymbol{y}$. Moreover, if we want $\frac{d\boldsymbol{y}}{d\boldsymbol{p}}(t_i)$ for each t_i, then we must apply the adjoint approach at each t_i. This is because we are equivalently asking for the sensitivity of a vector valued objective function at each t_i, which is not what the adjoint method is efficient at computing. As we discussed in section 2.1, the forward approach is better suited to this task, since it directly approximates the model sensitivities.

2.2.2 Case of LSQ Objective Function

We now make use of a different characterization of (4) that is consistent with the form of the objective function (6). We re-write $J(\boldsymbol{p}) \equiv G(\boldsymbol{y}(\boldsymbol{p}))$ using the Dirac-delta function as,

$$J(\boldsymbol{p}) = \int_0^T g(\boldsymbol{y}(t,\boldsymbol{p}))\,dt = \int_0^T \Big[\sum_{i=1}^{n_o}\sum_{j=1}^{n_y} \frac{(\tilde{\boldsymbol{y}}_j(t_i) - \boldsymbol{y}_j(t_i,\boldsymbol{p}))^2}{2}\delta(t_i - t)\Big]\,dt,$$

where $\boldsymbol{y}_j(t_i,\boldsymbol{p})$ is the j^{th} component of the solution of the ODE at time t_i and $\delta(t - t_i)$ is the Dirac-delta function, which is zero everywhere, except at $t = t_i$. $\delta(t - t_i)$ has the property that,

$$\int_a^b q(t)\delta(t - t_i)dt = q(t_i), \tag{15}$$

for any sufficiently smooth function $q(s)$, if $a < t_i < b$. With this representation for g,

$$g_{\boldsymbol{y}} = \sum_{i=1}^{n_o}(\tilde{\boldsymbol{y}}(t_i) - \boldsymbol{y}(t_i,\boldsymbol{p}))\delta(t - t_i).$$

In this case, we must appropriately handle the $\sum_i(\tilde{\boldsymbol{y}}(t_i) - \boldsymbol{y}(t_i,\boldsymbol{p}))\delta(t - t_i)$ term when solving the adjoint IVP. The presence of this term will lead to discontinuities in $\boldsymbol{\lambda}^T(t)$, whenever $t = t_i$. The natural way to account for these discontinuities is to solve (12) on each subinterval (t_i, t_{i+1}), and apply the jumps, to obtain the initial conditions, for the next subinterval (t_{i-1}, t_i),

$$\boldsymbol{\lambda}^T(t_i)^- = \boldsymbol{\lambda}^T(t_i)^+ + (\tilde{\boldsymbol{y}}(t_i) - \boldsymbol{y}(t_i,\boldsymbol{p})). \tag{16}$$

It is the cost of applying this jump condition at each t_i, that makes the cost of the adjoint approach particularly sensitive to the number of observations. We will now briefly review how the adjoint method extends to constant lag DDEs.

2.3 Adjoint Approach for constant lag DDEs

We consider here the special case of constant lag DDEs with a delay of the form $\alpha = t - \tau$ and constant history function, $\boldsymbol{y}(t) = \boldsymbol{y}_o$, for $t < 0$. For simplicity, we assume there is only one delay in the following derivation, but everything extends in a straightforward way to multiple delays. For a rigorous derivation of the adjoint method for more general systems of DDEs, we refer the reader to [15].

The difference here compared to the ODE adjoint system, (12), is that $\boldsymbol{f}$ not only depends on $\boldsymbol{y}(t)$, but also on $\boldsymbol{y}(t - \tau)$. For convenience, we let $\boldsymbol{\nu} = \boldsymbol{y}(t - \tau)$.

Let $\boldsymbol{\lambda}^T(t)$ be any vector valued function of dimension n_y, defined for $t \in [0, T+\tau]$. Similar to the ODE case, we define a perturbed objective function,

$$J(\boldsymbol{p}) = G(\boldsymbol{y}(\boldsymbol{p})) + \int_0^T \boldsymbol{\lambda}^T(t)\big(\dot{\boldsymbol{y}}(t,\boldsymbol{p}) - \boldsymbol{f}(t,\boldsymbol{y}(t),\boldsymbol{y}(t-\tau),\boldsymbol{p})\big)\,dt. \tag{17}$$

Taking the derivative with respect to the parameters, we obtain,

$$\begin{aligned}
J_{\boldsymbol{p}} &= \frac{dG}{d\boldsymbol{p}} + \int_0^T \boldsymbol{\lambda}^T(t)\left(\frac{d\dot{\boldsymbol{y}}}{d\boldsymbol{p}}(t) - \frac{\partial}{\partial \boldsymbol{p}}\Big[\boldsymbol{f}(t,\boldsymbol{y}(t,\boldsymbol{p}),\boldsymbol{y}(t-\tau,\boldsymbol{p})\Big]\right)dt \\
&= \frac{dG}{d\boldsymbol{p}} \\
&\quad + \int_0^T \boldsymbol{\lambda}^T(t)\Big(\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{\nu}}(t)\big(\boldsymbol{y}_{\boldsymbol{p}}(t-\tau) + \boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t)\big) - \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt \\
&= \int_0^T \Big(g_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) + \boldsymbol{\lambda}^T(t)\Big(\dot{\boldsymbol{y}}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{y}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t) - \boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t-\tau)\Big)\Big)\,dt \\
&\quad - \int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt.
\end{aligned}$$

In a similar way to the derivation of (10), we can re-write the above expression as,

$$\begin{aligned}
J_{\boldsymbol{p}} = \quad & \int_0^T \big(g_{\boldsymbol{y}}(t) + \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t) - \dot{\boldsymbol{\lambda}}^T(t)\big)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt - \int_0^T \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t-\tau)\,dt \\
& - \int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt - \Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T.
\end{aligned} \tag{18}$$

Now, after a change of variables and rewriting the second integral in (18) as,

$$\begin{aligned}
\int_0^T \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}_{\boldsymbol{p}}(t-\tau)\,dt &= \int_{-\tau}^{T-\tau} \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt \\
&= \int_0^T \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt \\
&\quad + \int_{-\tau}^0 \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt.
\end{aligned}$$

The second equality follows from being able to extend the integral to T by requiring that $\boldsymbol{\lambda}^T(t) = \boldsymbol{0}$ for $t \geq T$, and splitting the interval of integration. We now see that we can combine the first integral in this expression, with the first integral in (18), and after rearranging,

$$\begin{aligned} J_{\boldsymbol{p}} = &\int_0^T \big(-\dot{\boldsymbol{\lambda}}^T(t) + g_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\big)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt \\ &- \int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt \\ &+\Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T + \int_{-\tau}^0 \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt. \end{aligned}$$

The adjoint system for this constant lag DDE is defined by requiring that $\boldsymbol{\lambda}^T(t)$ be the solution of the IVP,

$$\dot{\boldsymbol{\lambda}}^T(t) = g_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t)\boldsymbol{f}_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\,;\boldsymbol{\lambda}^T(t) = \boldsymbol{0}, \text{ for } t \geq T. \quad (19)$$

As in the ODE case, this choice for $\boldsymbol{\lambda}^T(t)$ eliminates the integral involving $\boldsymbol{y}_{\boldsymbol{p}}(t)$ and results in the sensitivities being given by,

$$\begin{aligned} J_{\boldsymbol{p}} = &-\int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt \\ &+\Big(\boldsymbol{\lambda}^T(t)\boldsymbol{y}_{\boldsymbol{p}}(t)\Big)\Big|_0^T + \int_{-\tau}^0 \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{y}_{\boldsymbol{p}}(t)\,dt \\ J_{\boldsymbol{p}} = &-\int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f}_{\boldsymbol{p}}(t)\Big)\,dt \\ &-\boldsymbol{\lambda}^T(0)\boldsymbol{y}_{\boldsymbol{p}}(0) - \int_{-\tau}^0 \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)\boldsymbol{h}_{\boldsymbol{p}}(t)\,dt. \end{aligned}$$

Note that because the last integral is for $t \leq 0$, we have replaced $\boldsymbol{y}_{\boldsymbol{p}}(t)$ with $\boldsymbol{h}_{\boldsymbol{p}}(t)$ (recall that $\boldsymbol{h}$ is the history function).

2.3.1 Additional Considerations for DDEs

For DDEs, we have to be careful to properly handle discontinuities and storage of the solution on previous subintervals. As we did in the ODE case, we have to restart the solver at each observation point, t_i, when integrating the adjoint IVP from T to 0. However, the discontinuities introduced by (16) at each t_i will be encountered again, since $\dot{\boldsymbol{\lambda}}^T(t)$ depends on the lagged value, $\boldsymbol{\lambda}^T(t+\tau)$. Also, discontinuities in $\boldsymbol{y}(t)$ must be taken into consideration as well. It is usually the case that $\boldsymbol{y}(t)$ will be continuous, but its higher derivatives may not be. For example, if the history is constant, $h(t) = y_0$, then the derivative to the left of the initial time is zero, while it is $f(t, y_0, p)$ to the right of the initial time. For a fixed lag, $t - \tau$, we will encounter this discontinuity again when $t = \tau$. At this point, the discontinuity is propagated, but in a derivative of order one higher. At some point, this discontinuity is of sufficiently high order that it can be ignored. This will be the case when the order of the derivative discontinuity is higher than the order of the underlying CRK formula.

To handle these additional discontinuities, we should restart the solver at each of them. In Figure 1, we illustrate the impact that the discontinuities have on the behaviour of the solutions of a DDE model and its associated adjoint system. As we will see, the presence of these discontinuities will make the adjoint approach significantly more expensive for the case of DDEs than it is for ODEs.

For a detailed discussion of how discontinuities in DDEs impact the smoothness of the sensitivities, we refer the reader to [2].

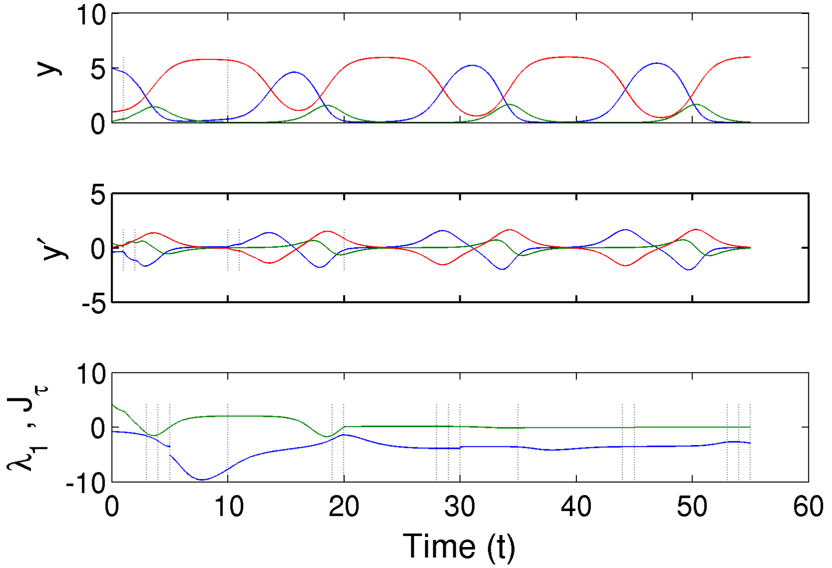

Fig. 1 Top: Solution of each component of the Kermack-McKendrick model (see 3.2.2). Middle: Derivative of each component of the solution of the model. Bottom: The adjoint variable for the first component of the model (blue), and the cumulative value of the integral, J_{τ_2} (green). The locations of discontinuities, up to second order, are indicated by dotted lines at the times where they occur. For this model, $\tau_1 = 1$, and $\tau_2 = 10$. Observations are at times 5,30, and 55.

3 Numerical Experiments

3.1 Implementation

For our experiments, we have implemented the adjoint method in the DDEM package, which already includes an implementation of the variational approach. DDEM uses an order 6 CRK IVP solver to maintain an order 6 piecewise polynomial approximation to the solution over the entire interval, $[0, T]$.

3.2 Experiments

In this section, we introduce two test problems, investigate how the adjoint method compares with the variational approach under various conditions, and discuss some key aspects of the adjoint approach. In each experiment, we generate observations by simulating the models very accurately with specified parameter values and then add normally distributed noise. The sensitivities are then computed for the specified parameter values.

3.2.1 Barnes Problem

The Barnes Problem is commonly used in the literature on parameter estimation for IVP models [13,12]. It refers to a specific parameterization of the following predator-prey model.

$$y_1{}'(t) = p_1 y_1(t) - p_2 y_1(t) y_2(t)$$

$$y_2{}'(t) = p_2 y_1(t) y_2(t) - p_3 y_2(t)$$

For our test problem, the parameters and initial conditions are chosen to be $p_1 = 1$, $p_2 = 1$, $p_3 = 1$, $y_1(0) = 1$, $y_2(0) = 0.3$, and $t \in [0, 20]$.

3.2.2 Kermack-McKendrick Model

This DDE system models the spread of disease within a population, where there are periodic outbreaks [6].

$$\begin{aligned} y_1{}'(t) &= -y_1(t) y_2(t - \tau_1) + y_2(t - \tau_2) \\ y_2{}'(t) &= y_1(t) y_2(t - \tau_1) - y_2(t) \\ y_3{}'(t) &= y_2(t) - y_2(t - \tau_2) \end{aligned}$$

For convenience, we denote the initial conditions as $\boldsymbol{y}(0) = [a, b, c]$. For our test problem, the parameters are chosen to be $a = 5$, $b = 0.1$, $c = 1$, $\tau_1 = 1$, $\tau_2 = 10$, and $t \in [0, 55]$.

3.2.3 Determining the Order of Discontinuities to Track in the adjoint DDEs

First, we investigate what order of discontinuity it is necessary to track, in order to obtain reasonable performance when approximating the adjoint DDE. Note that since we use reliable error control when approximating the adjoint IVP, even if we only track the jump discontinuities occurring at each of our observations, we should still obtain accurate sensitivities. This is illustrated in Table 1. However, we see that we end up either taking extra steps in order

Table 1 For several tolerances and maximum orders of discontinuity to track, we report the computer time taken by the adjoint method (including the time required for approximating the forward IVP) and the number of steps taken during the simulation of the adjoint IVP. This is done for the Kermack-McKendrick test problem, with $n_o = 5$.

Max Order	Tol	Max Rel Error	Time	Adjoint Steps
0	0.001	0.00704	0.0144	106
1	0.001	0.00678	0.0135	**101**
2	0.001	0.00705	0.0131	106
3	0.001	0.00702	0.0145	127
4	0.001	0.00695	0.0168	159
5	0.001	0.0077	0.0183	182
6	0.001	0.00733	0.0197	204
7	0.001	0.00764	0.0214	231
8	0.001	0.00748	0.023	257
0	0.0001	0.000236	0.0228	165
1	0.0001	0.000236	0.0179	135
2	0.0001	0.000236	0.0163	**131**
3	0.0001	0.000236	0.0165	144
4	0.0001	0.000236	0.0176	167
5	0.0001	0.000236	0.0194	191
6	0.0001	0.000236	0.0206	213
7	0.0001	0.000236	0.0223	239
8	0.0001	0.000236	0.024	265
0	1e-05	5.39e-05	0.0369	282
1	1e-05	5.39e-05	0.0223	182
2	1e-05	5.39e-05	0.0186	**163**
3	1e-05	5.39e-05	0.019	175
4	1e-05	5.39e-05	0.0206	200
5	1e-05	5.39e-05	0.0223	224
6	1e-05	5.39e-05	0.0233	243
7	1e-05	5.39e-05	0.0244	263
8	1e-05	5.39e-05	0.0255	283

to satisfy the error requirements or we take extra steps when we force the integration to stop at locations of all the identified discontinuities, which are not based on controlling the local truncation error of the adjoint IVP. Given these results, we only track discontinuities in the solution of the adjoint DDE and its first and second derivatives for the remainder of our experiments. (The resulting error control for the adjoint IVP will be less reliable but adequate for most problems.)

3.2.4 Dependence on n_o

We also investigate how many observations we can have before the cost of the adjoint approach is prohibitively expensive, relative to that associated with solving the variational equations. For the case of ODEs, we consider the Barnes problem. As we can see in Figure 2, the adjoint approach requires less computer time than the variational approach up to around 400 observations. We also note that for small numbers of observations, the adjoint method performs fairly

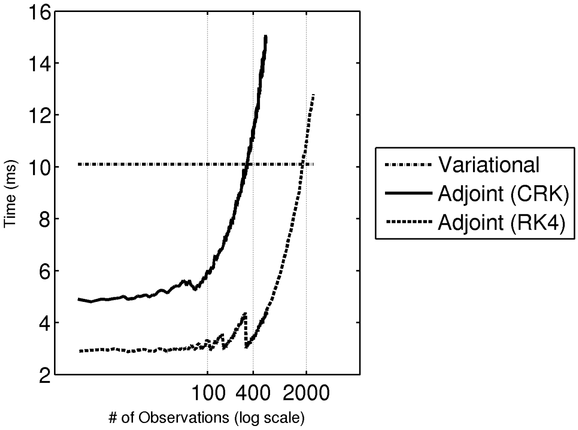

Fig. 2 For TOL $= 10^{-6}$, we plot the time taken by the adjoint and variational approaches versus the number of observations for the Barnes ODE. We also show how using a fourth order RK method with no error control for the adjoint ODE can reduce the computer time.

consistently, up until the spacing of the observations begins to restrict the step size the solver is able to take. For the variational approach, the cost remains flat, since increasing n_o only increases the number of off mesh interpolations we have to make in evaluating (5), which is much cheaper than the cost of simulating the variational equations.

For DDEs, we consider the Kermack-McKendrick model. As we see in Figure 3, the cost of the adjoint approach depends strongly on the number of observations. For example, with TOL $= 10^{-5}$, the adjoint method is already more expensive than the variational approach when there are more than 6 observations.

3.2.5 Low Order Method for Approximating the Adjoint Differential Equations

As discussed above, if we have a large number of observations, then our step size during solution of the adjoint equation might be severely restricted by having to restart the integration at each observation point rather than by only having to ensure the numerical accuracy of the solution be obtained. In such cases, it might be more efficient to use a lower order RK method for the adjoint ODEs or a lower order CRK method for the adjoint DDEs. As an example, for the ODE model, we have applied a fixed step size fourth order RK method (denoted RK4), with no associated error control, to the associated adjoint IVP. The impact on the cost is shown in Figure 2. We see that by using the lower order solver for the adjoint ODE, we are able to handle around 2400 observations before the adjoint method requires more computer time than the variational approach. On the other hand, the accuracy and reliability of the error in the approximate solution of the adjoint IVP will be reduced.

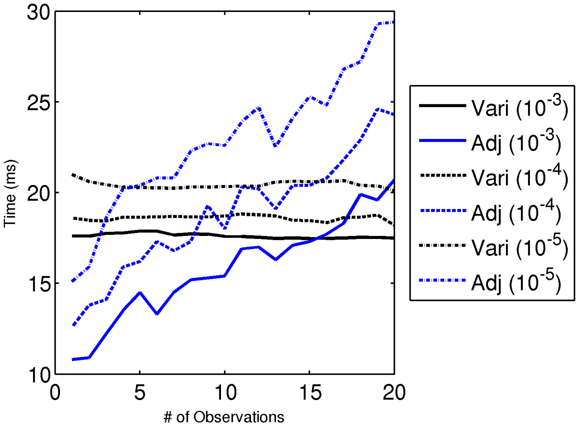

Fig. 3 For different tolerances, we plot the time taken by the adjoint (blue) and variational (black) approaches versus the number of observations for the Kermack-McKendrick DDE.

4 Discussion

4.1 Effect of the number of Delays in DDEs

Recall that the variational approach requires approximating the matrix valued IVP,

$$\frac{d}{dt}\boldsymbol{y_p}(t) = \boldsymbol{f_y}(t)\boldsymbol{y_p}(t) + \boldsymbol{f_p}(t) + \boldsymbol{f_\nu}(t)\boldsymbol{y_p}(t-\tau) - \boldsymbol{y}'(t-\tau)\tau_{\boldsymbol{p}}. \tag{20}$$

If we have more than one delay, $n_d > 1$, then $\boldsymbol{f_\nu}(t)\boldsymbol{y_p}(t-\tau) - \boldsymbol{y}'(t-\tau)\tau_{\boldsymbol{p}}$ on the RHS of (20) is replaced by,

$$\sum_{i=1}^{n_d} \boldsymbol{f}_{\boldsymbol{\nu}_i}(t)\boldsymbol{y_p}(t-\tau_i) - \boldsymbol{y}'(t-\tau_i)\tau_{i\boldsymbol{p}},$$

where $\boldsymbol{\nu}_i = \boldsymbol{y}(t-\tau_i)$. As such, we see that the cost of computing the RHS of (20) will have a linear dependence on the number of delays. For the adjoint approach, recall that the adjoint matrix valued IVP is given by,

$$\dot{\boldsymbol{\lambda}}^T(t) = g_{\boldsymbol{y}}(t) - \boldsymbol{\lambda}^T(t)\boldsymbol{f_y}(t) - \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f_\nu}(t+\tau)\,;\boldsymbol{\lambda}^T(t) = \mathbf{0}, \text{ for } t \geq T, \tag{21}$$

and the sensitivities are given by,

$$\begin{aligned} J_{\boldsymbol{p}} = &-\int_0^T \boldsymbol{\lambda}^T(t)\Big(\boldsymbol{f_\nu}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t) + \boldsymbol{f_p}(t)\Big)\,dt \\ &-\boldsymbol{\lambda}^T(0)\boldsymbol{y_p}(0) - \int_{-\tau}^0 \boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f_\nu}(t+\tau)\boldsymbol{h_p}(t)\,dt. \end{aligned} \tag{22}$$

For the case of multiple delays, we have that the term $\boldsymbol{\lambda}^T(t+\tau)\boldsymbol{f}_{\boldsymbol{\nu}}(t+\tau)$, on the RHS of both (21) and (22), is replaced by,

$$\sum_{i=1}^{n_d} \boldsymbol{\lambda}^T(t+\tau_i)\boldsymbol{f}_{\boldsymbol{\nu}_i}(t+\tau_i), \tag{23}$$

and $\boldsymbol{f}_{\boldsymbol{\nu}}(t)\boldsymbol{y}'(t-\tau)\alpha_{\boldsymbol{p}}(t)$ on the RHS of (22) is replaced by,

$$\sum_{i=1}^{n_d} \boldsymbol{f}_{\boldsymbol{\nu}_i}(t)\boldsymbol{y}'(t-\tau_i)\alpha_{i\boldsymbol{p}}(t). \tag{24}$$

For each τ_i, in order to evaluate $\boldsymbol{f}_{\boldsymbol{\nu}_i}(t+\tau_i)$ in (23), we require $\boldsymbol{y}(t+\tau_i-\tau_j)$, for each $j = 1, \ldots, n_d$. Hence, the cost of approximating (21) and (22) will require work proportional to the square of the number of delays, and the cost of evaluating (24) will be linear in the number of delays. Furthermore, we also have to restart the solver each time we encounter a discontinuity in $\boldsymbol{y}$, $\boldsymbol{\lambda}$, or any of their low order derivatives. This means that we have to restart each time $\boldsymbol{y}(t+\tau_i-\tau_j)$ (or one of its derivatives) has a discontinuity, as well as whenever $\boldsymbol{\lambda}^T(t+\tau_i)$ (or one of its derivatives) has a discontinuity.

4.2 Parallelism

Up to now, we have been assuming that we would be implementing the different approaches in a sequential computing environment. In real applications, we would like to exploit parallelism where possible. For both the adjoint and variational approaches, it is possible to divide the work based on subsets of the parameters and distribute the work amongst the available processors. Note that since both the variational and adjoint systems are linear (and possibly homogeneous) systems of differential equations, the principle of superposition applies and this observation can lead to opportunities to exploit parallelism. For the variational approach, this means that each column of $\boldsymbol{y}_{\boldsymbol{p}}$ can be approximated independently in parallel. For the adjoint approach, $\boldsymbol{\lambda}^T(t)$ is a vector, so we can not take advantage of superposition to efficiently approximate $\boldsymbol{\lambda}^T(t)$. However, once we have approximated $\boldsymbol{\lambda}^T(t)$, we can evaluate each component of $J_{\boldsymbol{p}}$ in parallel.

5 Conclusions and Future Work

We have discussed the variational and adjoint approaches for computing model sensitivities and sensitivities for a LSQ objective function. We identified how the cost of the adjoint approach can be much more sensitive to the number of observations than the variational approach and demonstrated how using a low order RK scheme for simulating the adjoint IVP can reduce the cost in the ODE case. We also discussed how, for DDEs, the cost of the variational approach scales linearly with the number of delays, while the cost scales with the square of the number of delays for the adjoint approach.

The DDEM package requires the user to provide code for computing the partials of the functions specifying the DDE system. Currently, we symbolically compute these partials in a pre-processing step, which provides us with the source code to compute them at runtime. Alternatively, we could use Automatic Differentiation [3] (AD) to compute the partials at runtime. In future, we will explore how this choice impacts performance.

We also plan to consider the case of larger scale models (where n_y is very large) and the adjoint approach is often used [10]. For the case of LSQ, we would like to see how many observations are needed before the adjoint approach becomes less effective for these problems. We also plan to further investigate and quantify the potential of exploiting parallelism in both the adjoint and variational approaches for problems that have special structure.

References

1. Alexe, M., Sandu, A.: Forward and adjoint sensitivity analysis with continuous explicit rungekutta schemes. Applied Mathematics and Computation **208**(2), 328 – 346 (2009)
2. Baker, C.T., Paul, C.A.: Pitfalls in parameter estimation for delay differential equations. SIAM Journal on Scientific Computing **18**(1), 305–314 (1997)
3. Bischof, C.H., Hovland, P.D., Norris, B.: On the implementation of automatic differentiation tools. Higher Order Symbol. Comput. **3**(21), 311–331 (2008)
4. Cao, Y., Li, S., Petzold, L., Serban, R.: Adjoint sensitivity analysis for differential algebraic equations: The adjoint dae system and its numerical solution. SIAM J. Sci. Comput. **3**(24), 1076–1089 (2003)
5. Hutchinson, G.E.: Circular causal systems in ecology. Annals of the New York Academy of Sciences **50**(4), 221–246 (1948)
6. Kermack, W., McKendrick, A.: A contribution to the mathematical theory of epidemics. In: Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences, vol. 115, pp. 700–721. The Royal Society (1927)
7. Kuang, Y.: Delay differential equations: with applications in population dynamics. Academic Press (1993)
8. Lenz, S., Schlder, J., Bock, H.: Numerical computation of derivatives in systems of delay differential equations. Mathematics and Computers in Simulation **96**, 124–156 (2014)
9. Petzold, L., Li, S., Cao, Y., Serban, R.: Sensitivity analysis of differential-algebraic equations and partial differential equations. Computers & chemical engineering **30**(10), 1553–1559 (2006)
10. Sengupta, B., Friston, K., Penny, W.: Efficient gradient computation for dynamical models. NeuroImage **98**, 521–527 (2014)
11. Shakeri, F., Dehghan, M.: Solution of delay differential equations via a homotopy perturbation method. Mathematical and Computer Modelling **48**(3), 486 – 498 (2008)
12. Varah, J.: A spline least squares method for numerical parameter estimation in differential equations. SIAM Journal on Scientific and Statistical Computing **3**(1), 28–46 (1982)
13. Wang, B.: Parameter estimation for odes using a cross-entropy approach. Master's thesis, University of Toronto (2012)
14. Zivari-Piran, H.: Efficient simulation, accurate sensitivity analysis and reliable parameter estimation for delay differential equations. Ph.D. thesis, Univeristy of Toronto (2009)
15. Zivari-Piran, H., Enright, W.: Accurate first-order sensitivity analysis for delay differential equations: Part ii: The adjoint approach. preprint, Department of Computer Science, University of Toronto (2009)

16. Zivari-Piran, H., Enright, W.H.: Accurate first-order sensitivity analysis for delay differential equations. SIAM Journal on Scientific Computing **34**(5), A2704–A2717 (2012)