Accurate First-Order Sensitivity Analysis for Delay Differential Equations: Part I: The Forward Approach

Hossein ZivariPiran[1] and Wayne H. Enright[2]
[1]Department of Mathematics and Statistics, York University
Toronto, ON, M3J 1P3, Canada
hzp@mathstat.yorku.ca
[2]Department of Computer Science, University of Toronto
Toronto, ON, M5S 3G4, Canada
enright@cs.toronto.edu

Abstract

In this paper, we derive an equation governing the dynamics of first-order forward sensitivities for a general system of parametric neutral delay differential equations (NDDEs). We also derive a formula which identifies the size of jumps that appear at discontinuity points when the sensitivity equations are integrated. The formula leads to an algorithm which can compute sensitivities for various types of parameters very accurately and efficiently.

1 Introduction

Sensitivity analysis is concerned with the study of the relationship between infinitesimal changes in model parameters and changes in model outputs. Sensitivity information can be used to estimate which parameters are most influential in affecting the behavior of the simulation. Such information is crucial for experimental design, data assimilation, reduction of complex nonlinear models, and evaluating optimization gradients and Jacobians in the setting of dynamic optimization and parameter estimation.

Sensitivity analysis also plays a very important role in dynamical systems. For example, when investigating periodic orbits, Lyapunov exponents, or other chaos indicators, and for general bifurcation analysis, computation of the sensitivities with respect to the initial conditions of the problem is a key component of the analysis (see [2] and references therein for more details).

There are two main approaches to sensitivity analysis: *forward sensitivity analysis* and *adjoint sensitivity method*. In this paper, we only discuss the forward approach for delay differential equations (DDEs). The adjoint method for DDEs will be discussed in a subsequent paper [15].

The parameters affecting a state variable or a mathematical function are usually distinguished by being grouped separately in the list of arguments. For a system with a state variable $y(t)$, we refer to the parameterized version of the state variable by $y(t; \mathbf{p})$, where $\mathbf{p}$ is a vector of parameters ($\mathbf{p} \in \mathbb{R}^{\mathcal{L}}$).

The (first-order) solution (forward) sensitivity with respect to the model parameter p_l is defined as the vector

$$s_l(t; \mathbf{p}) = \{\frac{\partial}{\partial p_l}\} y(t; \mathbf{p}), \quad (l = 1, \dots, \mathcal{L}). \tag{1}$$

The above definition can be considered for all continuous time models including those where $y(t; \mathbf{p})$ is defined by a system of differential equations.

The simplest way of calculating sensitivity coefficients is to use a finite difference approximation (also called *external differentiation*),

$$\{\frac{\partial}{\partial p_l}\} y(t; \mathbf{p}) \approx \frac{y(t; \mathbf{p} + \mathbf{e}_l \Delta p_l) - y(t; \mathbf{p})}{\Delta p_l}. \tag{2}$$

This technique is very easy to implement because it requires no extra code beyond the original model ODE solver, although it does require more applications of the underlying solver (one for each partial derivative approximation and one for each component of $y(t; \mathbf{p})$), with the same value of a chosen tolerance Tol. However, when computations are done in finite precision, the presence of rounding errors prevents the use of a very small perturbation, Δp_l, and it has been shown that with the best choice for Δp_l, the approximation is only accurate to $\mathcal{O}(\sqrt{Tol})$ [3].

In the so-called *internal differentiation* approach, the governing equations for the first order sensitivity coefficients (*variational equations*) are derived along with the required initial/boundary conditions. These equations are usually solved simultaneously with the original equations of the system using an appropriate differential equation solver. This approach has been studied in detail for ordinary differential equations (ODEs) including initial value problems (IVPs) ([6], [11], [5], [2]) and differential algebraic equations (DAEs) ([13], [7]).

Following the method of internal differentiation, Baker and Rihan [1] have studied the sensitivity analysis problem for the restricted class of parameterized DDEs defined by,

$$\begin{aligned} y'(t; \mathbf{p}) &= f(t, y(t; \mathbf{p}), y(t - \sigma(t; \mathbf{p})); \mathbf{p}), \quad \text{for } t \geq t_0(\mathbf{p}), \\ y(t; \mathbf{p}) &= \phi(t; \mathbf{p}), \quad \text{for } t \leq t_0(\mathbf{p}). \end{aligned} \tag{3}$$

They derive the governing equations and study the particular issues accompanying the numerical computation of sensitivities, which mostly originate from the discontinuous nature of DDEs. However, so far, nothing has been suggested to overcome those difficulties for a general system of DDEs.

In this paper we derive the governing equations for a general system of DDEs (Section 2.1). We also identify the discontinuity issue for sensitivities and introduce the treatment as an explicit algorithm (Section 2.2). In Section 3 we give some numerical experiments.

2 First-Order Forward Sensitivity Computation for General DDEs

We consider the general case of a system of state-dependent neutral delay differential equations (NDDEs),

$$\begin{aligned} y'(t) &= f(t, y(t), y(\alpha_1(t, y; \mathbf{p})), \ldots, y(\alpha_\nu(t, y; \mathbf{p})), \\ &\qquad y'(\alpha_{\nu+1}(t, y; \mathbf{p})), \ldots, y'(\alpha_\omega(t, y; \mathbf{p})); \mathbf{p}), \quad \text{for } t \geq t_0(\mathbf{p}), \\ y(t_0) &= y_0(\mathbf{p}), \\ y(t) &= \phi(t; \mathbf{p}), \quad \text{for } t < t_0(\mathbf{p}), \\ y'(t) &= \phi'(t; \mathbf{p}), \quad \text{for } t < t_0(\mathbf{p}), \end{aligned} \tag{4}$$

where y, f, and ϕ are $\mathcal{M}$-vector functions, $\mathbf{p}$ is an $\mathcal{L}$-vector of parameters, and $\alpha_k(t, y; \mathbf{p})$, $k = 1, \ldots, \{\nu + \omega\}$ are scalar functions.

2.1 The Governing Equations

The governing equations for the first-order sensitivity coefficients are derived by differentiation of (4) with respect to a selected model parameter p_l and applying the chain rule and Clairaut's theorem, yielding (in the vector form)

$$\begin{aligned} s_l'(t) = \frac{\partial f}{\partial y} s_l(t) &+ \sum_{k=1}^{\nu} \left[\frac{\partial f}{\partial y(\alpha_k)} \left(y'(\alpha_k) \left(\frac{\partial \alpha_k}{\partial y} s_l(t) + \frac{\partial \alpha_k}{\partial p_l} \right) + s_l(\alpha_k) \right) \right] \\ &+ \sum_{k=\nu+1}^{\nu+\omega} \left[\frac{\partial f}{\partial y'(\alpha_k)} \left(y''(\alpha_k) \left(\frac{\partial \alpha_k}{\partial y} s_l(t) + \frac{\partial \alpha_k}{\partial p_l} \right) + s_l'(\alpha_k) \right) \right] \\ &+ \frac{\partial f}{\partial p_l}, \end{aligned} \tag{5}$$

where $s_l(t)$ is the $\mathcal{M} \times 1$ sensitivity coefficient vector ($s_{il} \equiv \frac{\partial y_i}{\partial p_l}$), $\frac{\partial f}{\partial y}$ is the $\mathcal{M} \times \mathcal{M}$ Jacobian matrix ($[\frac{\partial f}{\partial y}]_{ij} \equiv \frac{\partial f_i}{\partial y_j}$), $\frac{\partial f}{\partial y(\alpha_k)}$ is the $\mathcal{M} \times \mathcal{M}$ delayed Jacobian matrix ($[\frac{\partial f}{\partial y(\alpha_k)}]_{ij} \equiv \frac{\partial f_i}{\partial y_j(\alpha_k)}$), $\frac{\partial \alpha_k}{\partial y}$ is a $1 \times \mathcal{M}$ row-vector of partial derivatives ($[\frac{\partial \alpha_k}{\partial y}]_{1j} \equiv \frac{\partial \alpha_k}{\partial y_j}$), $\frac{\partial \alpha_k}{\partial p_l}$ is a scalar, and $\frac{\partial f}{\partial p_l}$ is an $\mathcal{M} \times 1$ vector of partial derivatives ($[\frac{\partial f}{\partial p_l}]_{i1} \equiv \frac{\partial f_i}{\partial p_l}$).

To find the sensitivity coefficient vector s_l we solve the delay differential system (5) simultaneously with the system (4), with associated initial functions

$$\begin{aligned} s_l(t_0) &= \frac{\partial y_0(\mathbf{p})}{\partial p_l}, \\ s_l(t) &= \frac{\partial \phi(t;\mathbf{p})}{\partial p_l}, \quad \text{for} \ \ t < t_0(\mathbf{p}), \\ s_l'(t) &= \frac{\partial \phi'(t;\mathbf{p})}{\partial p_l}, \quad \text{for} \ \ t < t_0(\mathbf{p}). \end{aligned}$$

2.2 Handling C^0 Discontinuities in Sensitivities

The discontinuities that arise in simulation of DDEs can also propagate to the sensitivity coefficients and a similar treatment is required to perform reliable computations. However, when we integrate a standard system of DDEs (4), no discontinuity of zero order (i.e. a discontinuity in the solution values) can appear after the starting point. But for sensitivities we may have C^0 discontinuities, because the associated DDEs (5) are not valid at some points. These are the points where Clairaut's theorem is not applicable. At these points we may have C^0 jumps. It is the appearance of C^0 discontinuities that makes the task of computing sensitivities challenging. In this section we will describe the source of C^0 discontinuities and compute the size of the jumps at these points of discontinuity. The integration then can be restarted with new computed starting values for the sensitivity equations.

2.2.1 Barton's Formula for Hybrid ODE Systems

Tolsma and Barton [13] have considered extensions to the classical sensitivity theory that define the parametric sensitivity of discontinuous systems. Consider the general case where a transition is triggered by a zero crossing of an event function $g(t, y, y'; \mathbf{p})$ at the point λ, and let $y(\lambda^-)$, $y(\lambda^+)$ be the values of the state variables before and after the event. If the state transition is continuous

$$y(\lambda^+) = y(\lambda^-), \tag{6}$$

then differentiating both sides of this equation with respect to a parameter p_l and some rearrangement yields,

$$\frac{\partial y}{\partial p_l}(\lambda^+) = \frac{\partial y}{\partial p_l}(\lambda^-) + \left(y'(\lambda^-) - y'(\lambda^+)\right) \frac{d\lambda}{dp_l}, \tag{7}$$

where $\frac{d\lambda}{dp_l}$ represents the sensitivity of the event time with respect to the parameter p_l. To compute the value of $\frac{d\lambda}{dp_l}$, we can differentiate $g(t, y, y'; \mathbf{p}) = 0$

w.r.t. p_l and rearrange terms to obtain,

$$\frac{\partial g}{\partial y'}\left(\frac{\partial}{\partial t}\left(\frac{\partial y}{\partial p_l}\right)+y''\frac{d\lambda}{dp_l}\right)+\frac{\partial g}{\partial y}\left(\frac{\partial y}{\partial p_l}+y'\frac{d\lambda}{dp_l}\right)+\frac{\partial g}{\partial p_l}+\frac{\partial g}{\partial t}\frac{d\lambda}{dp_l}=0, \quad (8)$$

which is a linear equation w.r.t. $\frac{d\lambda}{dp_l}$. (Note that all functions and partial derivatives in (8) are evaluated at λ^-.)

2.2.2 DDEs as Discontinuous IVPs

In [16] we showed that DDEs can be considered as a special subclass of discontinuous IVPs. Here we briefly review this correspondence. We use a simple system of DDEs to avoid notational complications. The transformation for general DDEs is discussed in [16].

Consider a simple state-dependent retarded delay differential equation (RDDE) with a single delay defined by

$$\begin{aligned} y'(t) &= f(t, y(t), y(\alpha(t, y(t)))), \quad \text{for } t \geq t_0 \\ y(t_0) &= y_0, \\ y(t) &= \phi(t), \quad \text{for } t < t_0 \end{aligned} \quad (9)$$

where $f(t, y, v)$ is assumed to be sufficiently differentiable with respect to t, y and v.

With this assumption, the only discontinuities in the solution or its low order derivatives will be associated with the propagation of discontinuities introduced by the initial function or at the initial point.

Now assume that jumps in one of the derivatives of $y(t)$ with respect to t occur at the points

$$\Lambda \equiv \{\cdots < \lambda_{-2} < \lambda_{-1} < \lambda_0 = t_0 < \lambda_1 < \lambda_2 < \cdots\} \quad (10)$$

where $\lambda_j,\ j < 0$, are the locations of discontinuities in the initial function. Then, artificial event functions

$$g_i(t, y(t)) = \alpha(t, y(t)) - \lambda_i, \quad i = \ldots, -2, -1, 0, 1, 2, \ldots \quad (11)$$

can be defined accordingly and used to write the equation characterizing the propagation of a discontinuity to $\lambda_r,\ r \geq 1$,

$$\lambda_r = \min\{\lambda > \lambda_{r-1} \colon \lambda \text{ is a root of odd multiplicity of } g_i(t, y(t)),\ \ i \leq r-1\}. \quad (12)$$

In other words, $\lambda_r,\ r \geq 1$, is the leftmost discontinuity of all propagated discontinuities arising from $\{\ldots, \lambda_{-1}, \lambda_0, \lambda_1, \ldots, \lambda_{r-1}\}$ and lying in $(\lambda_{r-1}, +\infty)$. The roots of $g_i(t, y(t))$ with even multiplicity do not cause discontinuities and they do not need to be identified, since the delay argument, $\alpha(t, y(t))$, *crosses* a previous discontinuity point only for roots which have odd multiplicity.

Note that for the special case involving a single increasing delay argument and a smooth history function, $\phi(t)$; each discontinuity is caused by propagation from the most recent previous discontinuity point, namely,

$$\alpha(\lambda_r, y(\lambda_r)) = \lambda_{r-1}, \quad r \geq 1. \tag{13}$$

Using the explicit identification of all sources of non-smoothness, it is not hard to see that the solution of the system (9) also satisfies the following system of discontinuous IVPs,

$$\begin{aligned} y'(t) &= f_i(t, y(t)) = f(t, y(t), y_{[i]}(\alpha(t, y(t)))), \\ &\quad \text{for } \lambda_i \leq \alpha(t, y(t)) < \lambda_{i+1} \\ y(t_0) &= y_0, \end{aligned} \tag{14}$$

where

$$y_{[i]}(\alpha) = \begin{cases} y(\alpha), & \text{for } \lambda_i \leq \alpha < \lambda_{i+1} \\ \text{smooth extension from } [\lambda_i, \lambda_{i+1}), & \text{for } \alpha < \lambda_i \text{ or } \alpha \geq \lambda_{i+1}. \end{cases} \tag{15}$$

The value of $y_{[i]}(\alpha)$ outside $[\lambda_i, \lambda_{i+1})$ is not required to be defined for the solution of (14), as the right hand side of (14) switches if α goes outside this interval. On the other hand, the smooth extension referred to in (15) is defined and used only to facilitate the root-finding process associated with accurately locating the point λ_r.

Now, using (11), Equation (14) can be rewritten in the standard form for specifying a discontinuous system of IVPs as

$$\begin{aligned} y'(t) &= f_i(t, y(t)), \\ &\quad \text{for } g_i(t, y(t)) \geq 0 \text{ and } g_{i+1}(t, y(t)) < 0 \\ y(t_0) &= y_0. \end{aligned} \tag{16}$$

2.2.3 Adapting for jumps in DDEs

Consider the case where a state transition arises in DDEs triggered by the propagation of a discontinuity of the solution. Since discontinuity points will, in general, depend on the value of parameters, we can define,

$$\Lambda(\mathbf{p}) \equiv \{\cdots < \lambda_{-2}(\mathbf{p}) < \lambda_{-1}(\mathbf{p}) < \lambda_0(\mathbf{p}) = t_0(\mathbf{p}) < \lambda_1(\mathbf{p}) < \lambda_2(\mathbf{p}) < \cdots\}, \tag{17}$$

which is the parameterized variant of (10). Considering the parameterized variant of event functions (Equation (11)), and letting i denote the index of the chosen minimum of Equation (12), $g_i(t, y; \mathbf{p})$ triggers a transition of the state variables at λ_r. Then, Equation (8) (using the fact that $\frac{\partial g_i}{\partial y'} = 0$), reduces to

$$\frac{\partial g_i}{\partial y}\frac{\partial y}{\partial p_l} + \frac{\partial g_i}{\partial p_l} + \left[\frac{\partial g_i}{\partial y}y' + \frac{\partial g_i}{\partial t}\right]\frac{d\lambda_r(\mathbf{p})}{dp_l} = 0. \tag{18}$$

For the partial derivatives we have the relations

$$\frac{\partial g_i}{\partial y} = \frac{\partial \alpha}{\partial y}, \tag{19}$$

$$\frac{\partial g_i}{\partial p_l} = \frac{\partial \alpha}{\partial p_l} - \frac{d\lambda_i(\mathbf{p})}{dp_l}, \tag{20}$$

$$\frac{\partial g_i}{\partial t} = \frac{\partial \alpha}{\partial t}. \tag{21}$$

Substituting into (18), we obtain

$$\frac{\partial \alpha}{\partial y}\frac{\partial y}{\partial p_l} + \frac{\partial \alpha}{\partial p_l} - \frac{d\lambda_i(\mathbf{p})}{dp_l} + \left[\frac{\partial \alpha}{\partial y}y' + \frac{\partial \alpha}{\partial t}\right]\frac{d\lambda_r(\mathbf{p})}{dp_l} = 0. \tag{22}$$

Assuming that $\frac{\partial \alpha}{\partial y}y' + \frac{\partial \alpha}{\partial t} \neq 0$, we are able to solve this linear equation to get

$$\frac{d\lambda_r(\mathbf{p})}{dp_l} = -\frac{\frac{\partial \alpha}{\partial y}(\lambda_r^-)\frac{\partial y}{\partial p_l}(\lambda_r^-) + \frac{\partial \alpha}{\partial p_l}(\lambda_r^-) - \frac{d\lambda_i(\mathbf{p})}{dp_l}}{\frac{\partial \alpha}{\partial y}(\lambda_r^-)y'(\lambda_r^-) + \frac{\partial \alpha}{\partial t}(\lambda_r^-)}, \tag{23}$$

and for the first discontinuity point ($\lambda_0(\mathbf{p}) = t_0(\mathbf{p})$) we have

$$\frac{d\lambda_0(\mathbf{p})}{dp_l} = \frac{\partial t_0(\mathbf{p})}{\partial p_l}, \tag{24}$$

and for the discontinuities in the history,

$$\frac{d\lambda_r(\mathbf{p})}{dp_l} = \frac{\partial \lambda_r(\mathbf{p})}{\partial p_l}, \qquad r = \ldots, -2, -1, \tag{25}$$

are independently computable, since $\lambda_r(\mathbf{p})$ is given as an input function for $r = \ldots, -2, -1$.

Equation (7) for DDEs becomes

$$\frac{\partial y}{\partial p_l}(\lambda_r^+) = \frac{\partial y}{\partial p_l}(\lambda_r^-) + \left(y'(\lambda_r^-) - y'(\lambda_r^+)\right)\frac{d\lambda_r(\mathbf{p})}{dp_l}, \tag{26}$$

and the steps for integrating the first-order sensitivity equations can be determined by the following algorithm

Algorithm 1: Computing Forward First-Order Sensitivities for DDEs

input : a general DDE (4); an approach for deriving and integrating the sensitivity equations with discontinuity tracking capability.

output: first-order sensitivity coefficients.

1.1 Initialize ($\lambda_0 = t_0(\mathbf{p})$).

1.2 $r \leftarrow 1$.

1.3 Integrate the equations up to a C^1 discontinuity point (λ_r).

1.4 Update the state variables (sensitivities) using

$$\frac{\partial y}{\partial p_l}(\lambda_r^+) = \frac{\partial y}{\partial p_l}(\lambda_r^-) + \left(y'(\lambda_r^-) - y'(\lambda_r^+)\right) \frac{d\lambda_r(\mathbf{p})}{dp_l}, \qquad (l = 1, \dots \mathcal{L}).$$

1.5 $r \leftarrow r + 1$ and restart (step **1.3**).

2.2.4 Computing $y''(\alpha_k)$

The term $y''(\alpha_k)$ in (5), needed for state-dependent and parameter-dependent NDDEs, can be computed using a similar technique. After differentiating (4) with respect to t, with $x(t) = y'(t)$ and some rearrangements, we obtain

$$\begin{aligned} x'(t) = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}x(t) + \sum_{k=1}^{\nu} \frac{\partial f}{\partial y(\alpha_k)} x(\alpha_k) \left(\frac{\partial \alpha_k}{\partial y}x(t) + \frac{\partial \alpha_k}{\partial t}\right) \\ + \sum_{k=\nu+1}^{\nu+\omega} \frac{\partial f}{\partial y'(\alpha_k)} x'(\alpha_k) \left(\frac{\partial \alpha_k}{\partial y}x(t) + \frac{\partial \alpha_k}{\partial t}\right). \end{aligned} \tag{27}$$

The associated initial functions are

$$\begin{aligned} x(t_0) &= y'(t_0), \\ x(t) &= \phi'(t; \mathbf{p}), \quad \text{for } \ t < t_0(\mathbf{p}), \\ x'(t) &= \phi''(t; \mathbf{p}), \quad \text{for } \ t < t_0(\mathbf{p}). \end{aligned}$$

Equation (27), when required, is integrated simultaneously with the other equations, providing the required values of $y''(\alpha_k) = x'(\alpha_k)$.

2.2.5 Handling Jumps in y''

Choosing $x(t) = y'(t)$ as a new state variable and integrating using the driven differential equations works well if $x(t)$ has no C^0 discontinuities (i.e. discontinuities in the value) after the starting time t_0. Otherwise, these jumps in the value cannot be captured by integrating using differential equations. In this case, which is inevitable when the original system is a system of NDDEs, these jumps should be treated as discrete events. Each time an event of this type is triggered, the initial values for the continued integration must be updated using the relation $x(\lambda^+) = y'(\lambda^+)$.

3 Numerical Results

3.1 Test Cases

The following cases are used to show the effectiveness of our approach. "Test Case 1" and "Test Case 3" are interesting situations where the sensitivity of the solution is observed with respect to parameters controlling the status of the starting point. This situation cannot be handled using traditional approaches. "Test Case 2" is a two-dimensional model with several parameters, including the parameters defining the components of the history function. "Test Case 4" is chosen to study the sensitivities for a system with chaotic behavior.

Test Case 1 Sensitivity of the solution with respect to a discontinuity at the initial point for [10],

$$y' = y(y(t)),$$

for t in $[2, 5.5]$. The history function is

$$y = 0.5, \quad \text{for } t < 2,$$

and

$$y(2) = 1.$$

The C^0 discontinuity of the solution at $\xi_0 = 2$ introduces break points at $\xi_1 = 4$ (C^1) and $\xi_2 = 4 + 2\ln 2 \approx 5.386$ (C^2).

The exact solution to this problem is

$$y(t) = \begin{cases} t/2, & \text{for } \xi_0 \leq t \leq \xi_1, \\ 2\exp(t/2 - 2), & \text{for } \xi_1 \leq t \leq \xi_2, \\ 4 - 2\ln(1 + \xi_2 - t) & \text{for } \xi_2 \leq t \leq 5.5. \end{cases}$$

The parameter is,

$$\mathbf{p} = [y(2)].$$

Test Case 2 Sensitivity of the solution with respect to all parameters and histories for a neutral delay logistic Gause-type predator-prey system [4],

$$y_1'(t) = y_1(t)(1 - y_1(t-\tau) - \rho y_1'(t-\tau)) - \frac{y_2(t) y_1(t)^2}{y_1(t)^2 + 1},$$

$$y_2'(t) = y_2(t)\left(\frac{y_1(t)^2}{y_1(t)^2 + 1} - \alpha\right),$$

where $\alpha = 1/10$, $\rho = 29/10$ and $\tau = 21/50$, for t in $[0, 30]$. The history functions are

$$\phi_1(t) = \frac{33}{100} - \frac{1}{10}t,$$
$$\phi_2(t) = \frac{111}{50} + \frac{1}{10}t,$$

for $t \leq 0$. The solution is C^1 discontinuous at the starting point which propagates as C^1 and C^2 discontinuities to $y_1(t)$ and $y_2(t)$, respectively, at $t = n\tau$ for $n \geq 1$.
The exact solution of this problem is unknown.

The parameters are,

$$\mathbf{p} = [\tau,\ \rho,\ \alpha,\ \mathrm{a},\ \mathrm{b},\ \mathrm{c},\ \mathrm{d}],$$

where

$$\phi_1(t) = \mathrm{a} + \mathrm{b}\,t,$$
$$\phi_2(t) = \mathrm{c} + \mathrm{d}\,t.$$

Test Case 3 Sensitivity of the solution with respect to the starting time for [9],

$$y'(t) = \frac{y(t)y(\ln(y(t)))}{t},$$

for t in $[1, 10]$. The history function is

$$\phi(t) = 1, \quad \text{for} \quad t \leq 1.$$

The exact solution to this problem is

$$y(t) = \begin{cases} t, & \text{for } 1 \leq t \leq e, \\ \exp(t/e), & \text{for } e \leq t \leq e^2, \\ \left(\frac{e}{3-\ln(t)}\right)^e, & \text{for } e^2 \leq t \leq e_3, \\ \text{not known}, & \text{for } e_3 < t, \end{cases}$$

where $e_3 = \exp(3 - \exp(1 - e))$.
Derivative jump discontinuities occur at $t = 1$ (C^1), $t = e$ (C^2), $t = e^2$ (C^3) and $t = e_3$ (C^4).

The parameter is,

$$\mathbf{p} = [t_0].$$

Test Case 4 Sensitivity of the solution with respect to the delay, exponent and history for a scalar equation that exhibits chaotic behavior. It is an example of the well known Mackey-Glass delay differential equations which they proposed as a model for the production of white blood cells

[8]. The problem has a constant delay and a constant history, and is defined by

$$y'(t) = \frac{2y(t-2)}{1+y(t-2)^{9.65}} - y(t),$$

for t in $[0, 100]$. The history function is

$$\phi(t) = 0.5, \quad \text{for } t \leq 0.$$

The exact solution of this problem is unknown.

The parameters are,

$$\mathbf{p} = [\tau,\ n,\ \mathrm{A}],$$

where

$$y'(t) = \frac{2y(t-\tau)}{1+y(t-\tau)^{n}} - y(t),$$

and

$$\phi(t) = \mathrm{A}, \quad \text{for } t \leq 0.$$

3.2 Results and Discussion

Figures 1–9 present the numerical results when we applied our approach to four test cases. Some interesting observations are made for each case in the respective captions. The results are also compared with the finite difference approach by showing the absolute error in the computed sensitivities for various parameter perturbations ($\Delta\mathbf{p}$). For this, we have used the results from our sensitivity analyzer code with a very tight tolerance (10^{-11}) as the exact values. The same tolerance (10^{-11}) was used for the simulations required in the finite difference approach. We also report in Table 1 the performance and accuracy of our code for "Test Case 1" for different tolerances.

This relationship between observed accuracy and the specified tolerance is typical of that obtained for all our test cases. It shows that for visualization purposes our approach delivers accurate approximations to the sensitivities at a very low cost. Therefore, acceptable visualizations can be computed using relaxed value of tolerance. In the plots of sensitivities we have used a default tolerance of 10^{-6}.

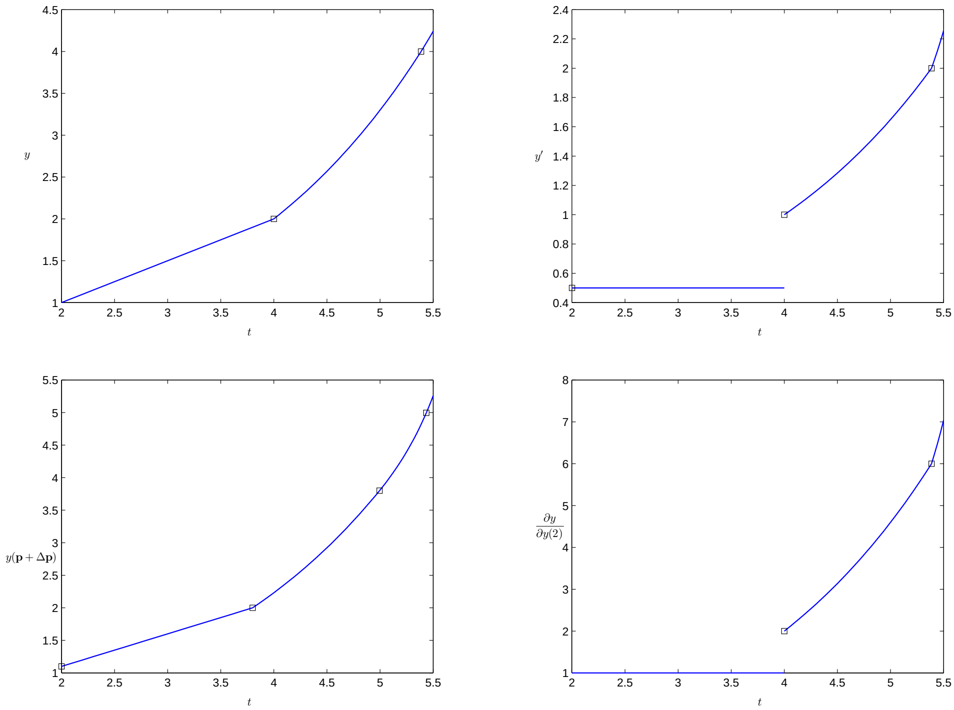

Figure 1: Plots of the numerical solution and the sensitivity for Test Case 1. Discontinuities of the solution at $t = 2, 4$ produces jumps in the sensitivity at those points. (Note that $y(\mathbf{p} + \Delta\mathbf{p})$ with $\Delta\mathbf{p} = 0.01$, has one more discontinuity point than y in the integration interval.)

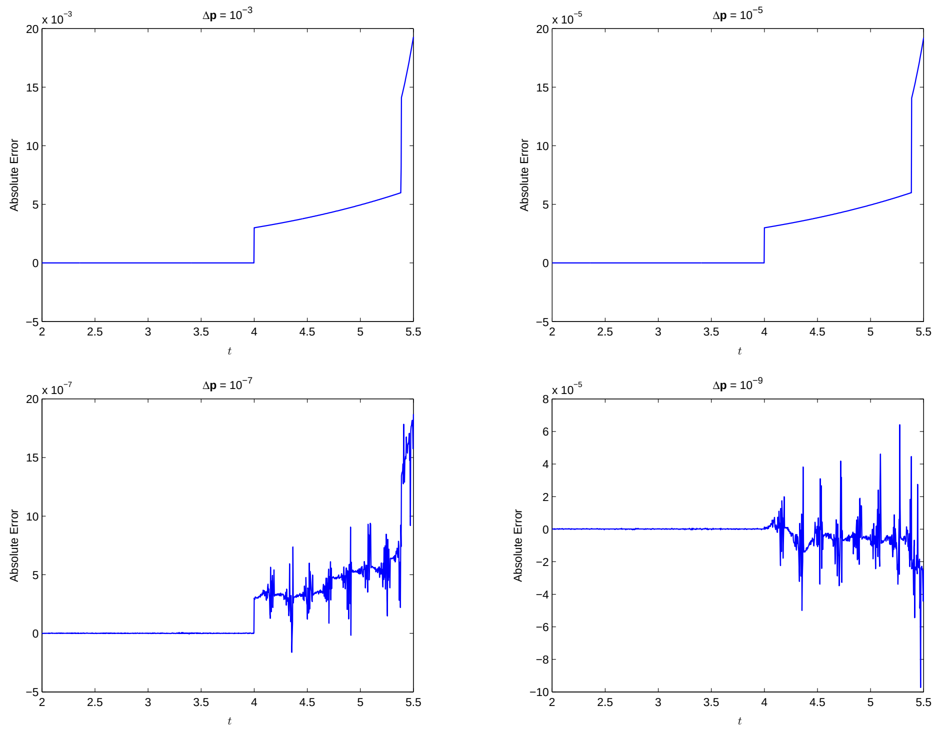

Figure 2: Plots of absolute errors of the sensitivity $\frac{\partial y}{\partial y(2)}$ computed using finite differences for Test Case 1. The limited accuracy of finite differences is clearly visible for $\mathbf{\Delta p} = 10^{-9}$.

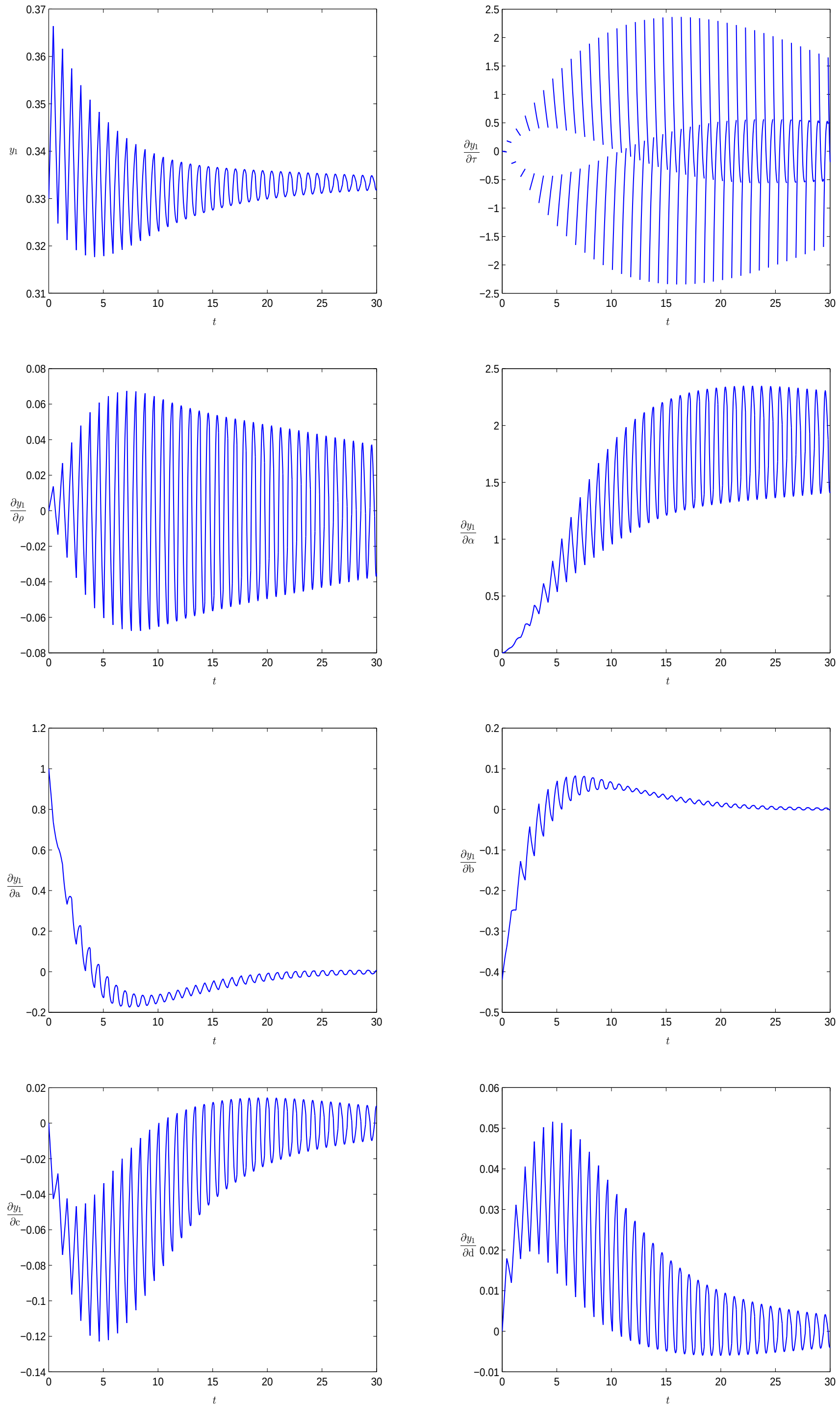

Figure 3: Plots of the numerical solution and the sensitivities for Test Case 2 (y_1). Discontinuities of the solution produce jumps in the sensitivity. Sensitivity coefficients for history related parameters, clearly show the transient effect of the history and the approximate time of this fading behavior.

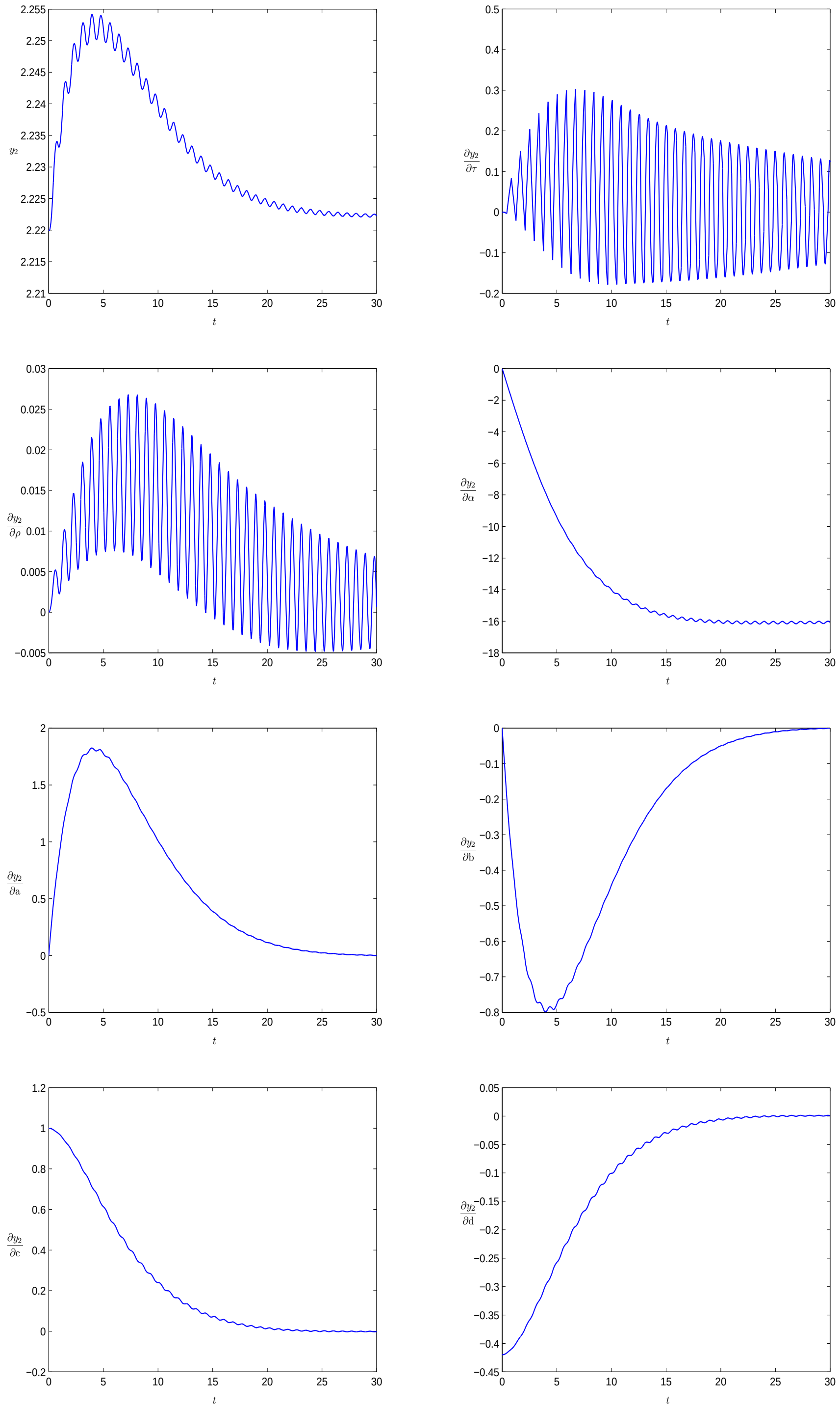

Figure 4: Plots of the numerical solution and the sensitivities for Test Case 2 (y_2). The sensitivity coefficient for the delay τ is smoother (persistent C^1 discontinuities), as well as the function itself (persistent C^2 discontinuities), compared to those of (y_1).

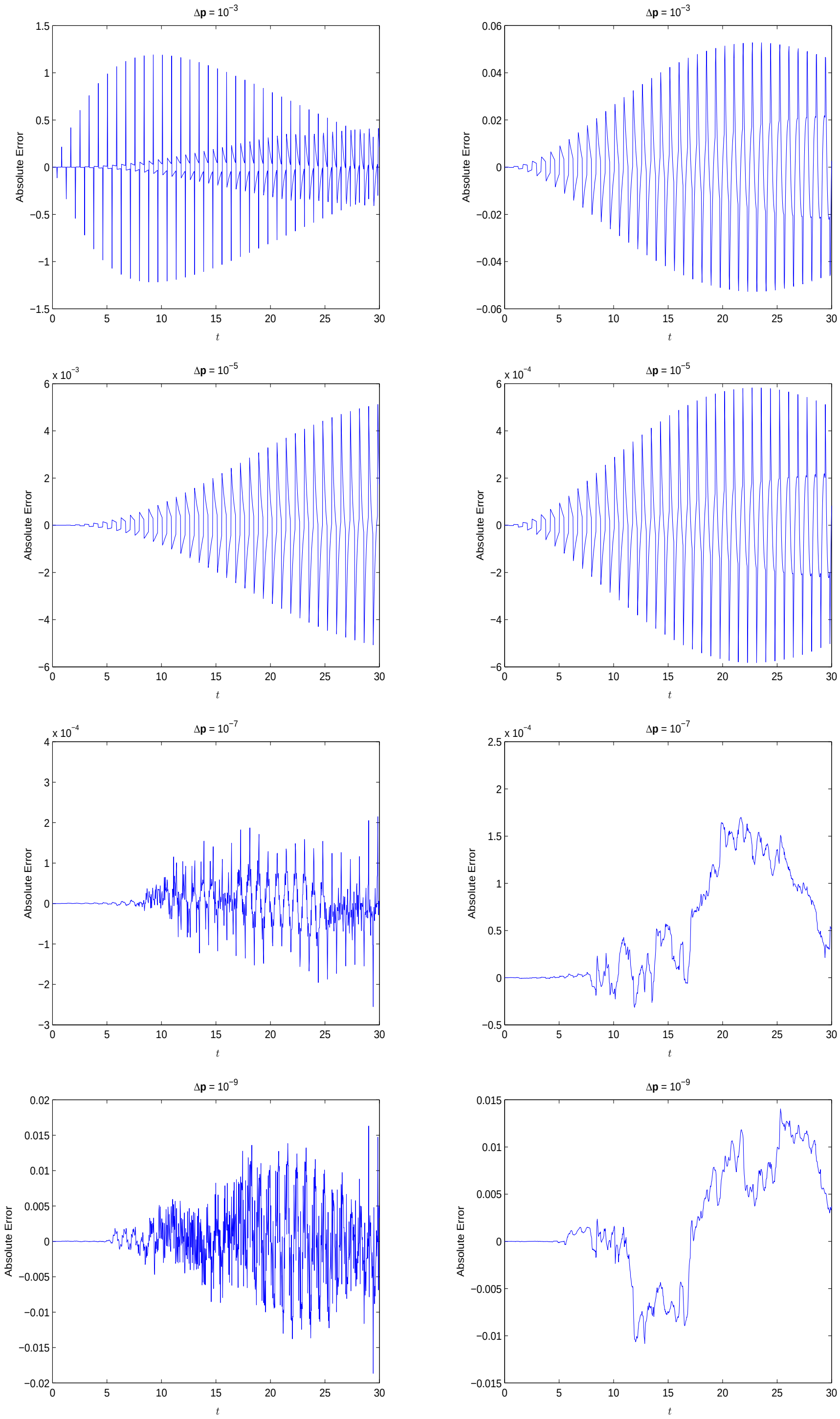

Figure 5: Plots of absolute errors of sensitivities $\frac{\partial y_1}{\partial \tau}$ (left column) and $\frac{\partial y_2}{\partial \tau}$ (right column) computed using finite differences for Test Case 2. The poor accuracy for $\mathbf{\Delta p} = 10^{-3}$ and the limited accuracy for $\mathbf{\Delta p} = 10^{-9}$ are clearly visible.

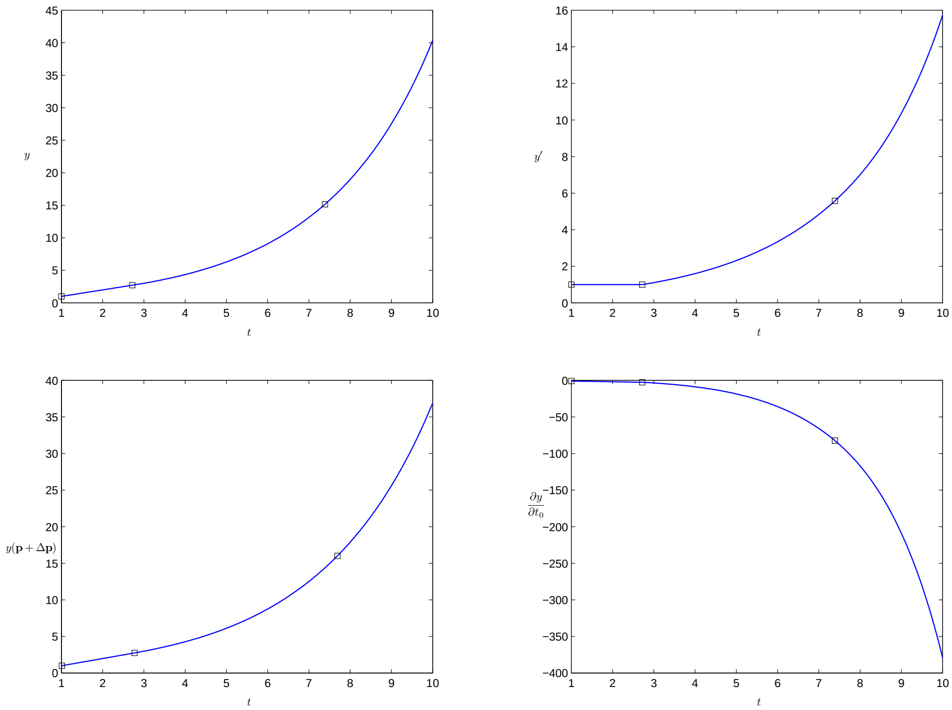

Figure 6: Plots of the numerical solution and the sensitivities for Test Case 3. The discontinuity of the solution at the starting point produces a jump in the sensitivity at that point (from 0 for $t < 1$, to -1 for $t = 1$). $y(\mathbf{p}+\Delta\mathbf{p})$ ($\Delta\mathbf{p} = 0.01$) shows a big reduction for a small change in the parameter as the sensitivity function predicts ($\frac{\partial y}{\partial t_0} \lll 0$).

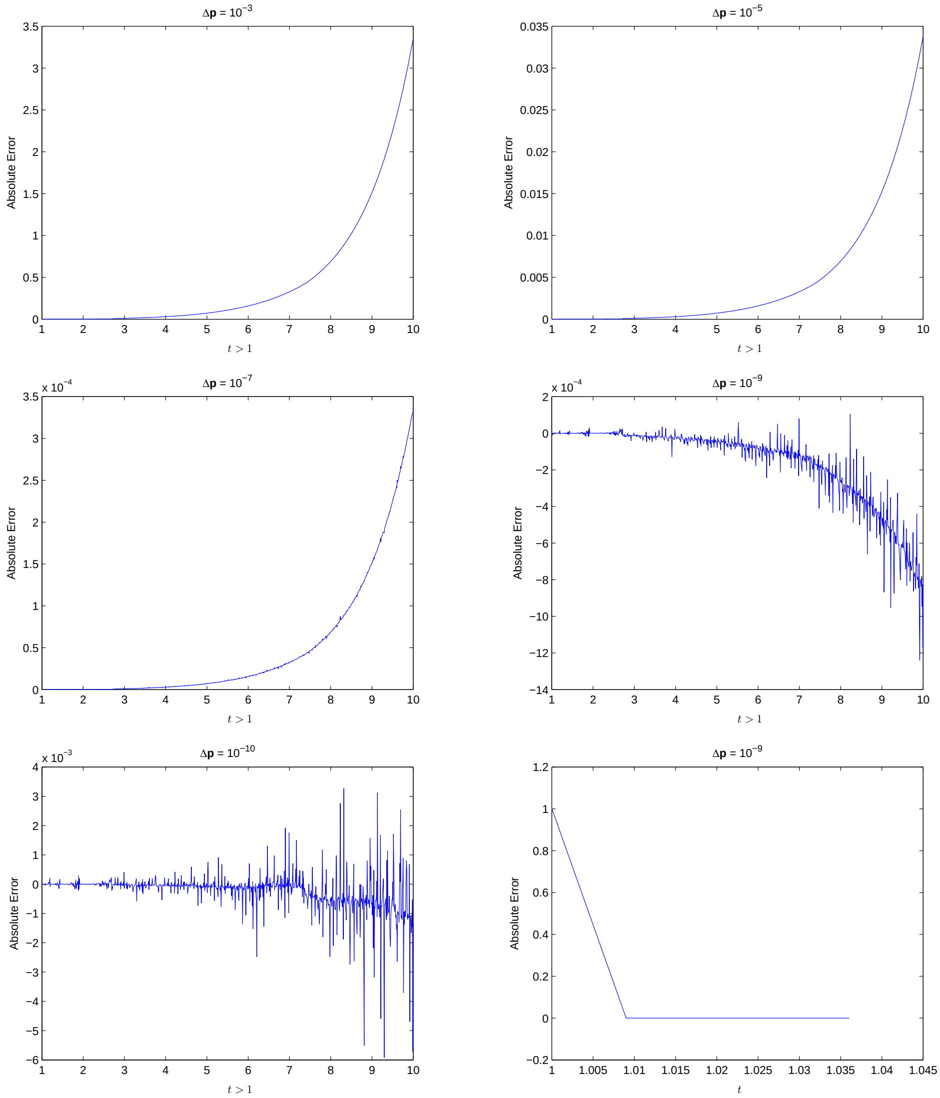

Figure 7: Plots of absolute errors of the sensitivity $\frac{\partial y}{\partial t_0}$ computed using finite differences for Test Case 3. The limited accuracy of finite differences is clearly visible for $\mathbf{\Delta p} = 10^{-10}$. Since the value of sensitivity computed by finite difference is very inaccurate at $t = t_0 = 1$ and the error is large, it is shown in a separate graph and excluded from the others.

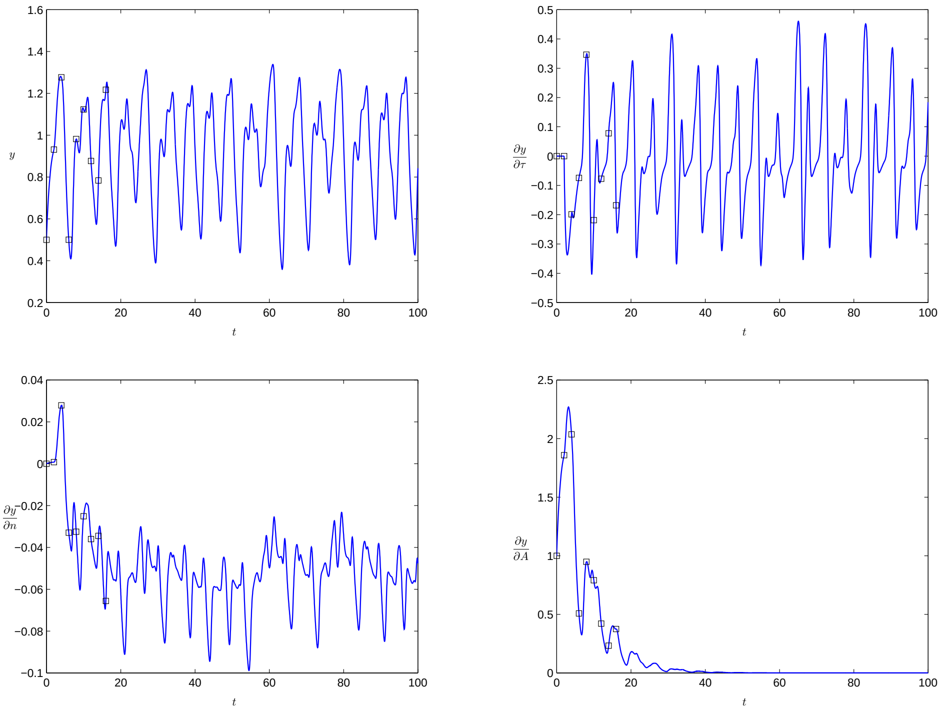

Figure 8: Plot of the numerical solution and the sensitivities for Test Case 4. The chaotic sensitivities $\frac{\partial y}{\partial \tau}$, $\frac{\partial y}{\partial n}$ and the non-chaotic sensitivity $\frac{\partial y}{\partial A}$ indicate that the chaos in y is a combined result of having a delay (τ) and an exponent (n) and is insensitive to the history.

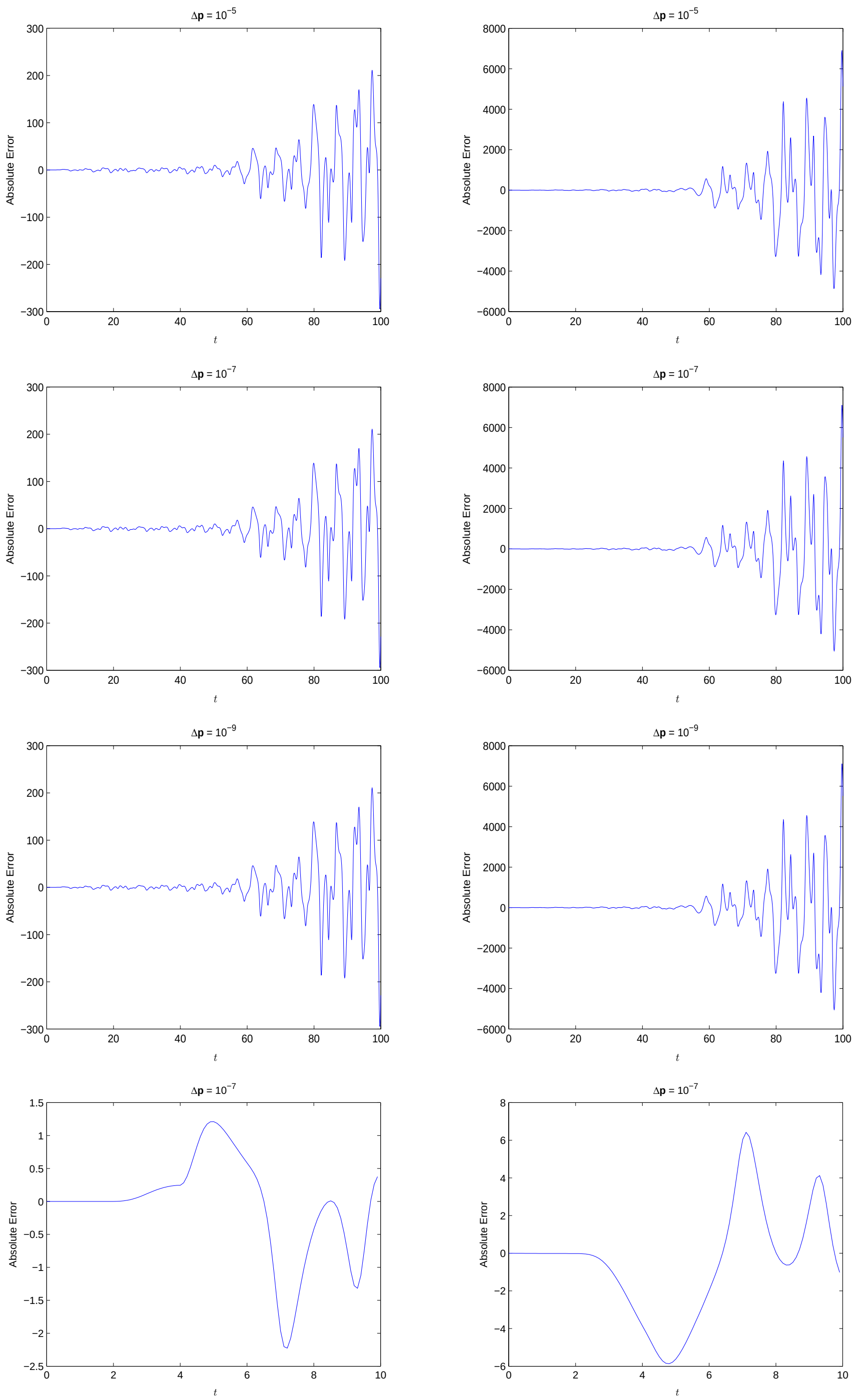

Figure 9: Plots of absolute errors of sensitivities $\frac{\partial y}{\partial \tau}$ (left column) and $\frac{\partial y}{\partial A}$ (right column) computed using finite differences for Test Case 4. The errors are extremely large; even for values near the starting point, similar large errors (more than 100%) can be seen if we look closely (bottom plots).

TOL	STEPS	REJECTS	FCN	ERROR
10^{-3}	6	0	69	$1.0 \cdot 10^{-6}$
10^{-5}	7	0	80	$1.1 \cdot 10^{-7}$
10^{-7}	10	1	124	$6.9 \cdot 10^{-8}$
10^{-9}	15	2	190	$5.2 \cdot 10^{-9}$
10^{-11}	24	3	300	-

Table 1: Summary Statistics of Sensitivity Analyzer for Test Case 1 with different tolerances (absolute tolerance = relative tolerance = TOL). The output with TOL = 10^{-11} is used as the exact value for error calculations. The reported error is the maximum absolute error in the sensitivities over the integration interval using 1000 equally spaced points.

4 Conclusions

We have developed and implemented an approach that determines accurate and reliable approximations to the first-order sensitivities of the solution of a system of DDEs. The approach can be applied to any numerical DDE method with discontinuity location capability (such as those discussed in [16] or [12]). It is shown that (as the specified tolerance, TOL, goes to zero) the max error in the sensitivities will be bounded by a small multiple of TOL. We know of no other technique for approximating sensitivities that can be as accurate. These accurate sensitivities have been used as the key component of a parameter determination technique that has been analyzed and implemented in [14]. The implemented sensitivity analyzer is part of the package DDEM which is available at the URL 'http://www.cs.toronto.edu/~hzp'.

References

[1] C.T.H. Baker and F.A. Rihan. Sensitivity Analysis of Parameters in Modelling with Delay-Differential Equations. Technical Report 349, Department of Mathematics, University of Manchester, Manchester, England, September 1999.

[2] R. Barrio. Sensitivity Analysis of ODEs/DAEs Using The Taylor Series Method. *SIAM J. Sci. Comput.*, 27(6):1929–1947, 2006.

[3] E. Hairer, S.P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I. Nonstiff problems.* Springer Series in Computational Mathematics. Springer-Verlag, New York, second edition, 1993.

[4] Y. Kuang. On neutral delay logistic Gause-type predator-prey systems. *Dynamics and Stability of Systems*, 6:173–189, 1991.

[5] S.L. Lee and P. D. Hovland. Sensitivity Analysis Using Parallel ODE Solvers and Automatic Differentiation in C: SensPVODE and ADIC. Technical Report ANL/MCS-P818-0500, Argonne National Laboratory, May 2000.

[6] J.R. Leis and M.A. Kramer. The Simultaneous Solution and Sensitivity Analysis of Systems Described by Ordinary Differential Equations. *ACM Trans. Math. Soft.*, 14(1):45–60, 1988.

[7] S. Li and L. Petzold. Software and algorithms for sensitivity analysis of large-scale differential algebraic systems. *Journal of Computational and Applied Mathematics*, 125:131–145, 2000.

[8] M.C. Mackey and L. Glass. Oscillation and Chaos in Physiological Control Systems. *Science*, 197(4300):287–289, 1977.

[9] K.W. Neves. Automatic Integration of Functional Differential Equations: An Approach. *ACM Trans. Math. Soft.*, 1(4):357–368, 1975.

[10] C.A.H. Paul. Developing a delay differential equation solver. *Appl. Numer. Math.*, 9:403–414, 1992.

[11] R. Serban and A.C. Hindmarsh. CVODES: An ODE Solver with Sensitivity Analysis Capabilities. Technical Report UCRL-JP-200039, Lawrence Livermore National Laboratory, September 2003.

[12] S. Thompson and L.F. Shampine. A Friendly Fortran DDE Solver. *Appl. Numer. Math.*, 53(3):503–516, 2006.

[13] J.E. Tolsma and P. Barton. Hidden Discontinuities and Parametric Sensitivity Calculations. *SIAM J. Sci. Comput.*, 23(6):1861–1874, 2002.

[14] H. Zivaripiran. *Efficient Simulation, Accurate Sensitivity Analysis and Reliable Parameter Estimation for Delay Differential Equations.* PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada, 2009.

[15] H. Zivaripiran and W.H. Enright. Accurate First-Order Sensitivity Analysis for Delay Differential Equations: Part II: The Adjoint Approach. preprint, Department of Computer Science, University of Toronto, 2009.

[16] H. Zivaripiran and W.H. Enright. An Efficient Unified Approach for the Numerical Solution of Delay Differential Equations. *Numerical Algorithms*, 53(2):397–417, 2010.