SCI INSTITUTE
TECHNICAL REPORT

Defect Sampling in Global Error Estimation for ODEs and Method-Of-Lines PDEs Using Adjoint Methods

Lethuy Tran and Martin Berzins

UUSCI-2011-006

Scientific Computing and Imaging Institute
University of Utah
Salt Lake City, UT 84112 USA

December 15, 2011

Abstract:

The importance of good estimates of the defect in the numerical solution of initial value problem ordinary differential equations is considered in the context of global error estimation by using adjoint-equation based methods. In the case of solvers based on the fixed leading coefficient backward differentiation formulae, the quality of defect estimates is shown to play a major role in the reliability of the global error estimator of Cao and Petzold. New defect estimates obtained by sampling the defect are derived to improve the quality and efficiency of adjoint-based global error estimation. The inclusion of only one estimate of the defect per timestep is shown to provide a good compromise between accuracy and efficiency for global error estimation of odes and method-of-lines pdes.

THE
UNIVERSITY
OF UTAH

DEFECT SAMPLING IN GLOBAL ERROR ESTIMATION FOR ODES AND METHOD-OF-LINES PDES USING ADJOINT METHODS [*]

LETHUY TRAN[†] AND MARTIN BERZINS[‡]

Abstract. The importance of good estimates of the defect in the numerical solution of initial value problem ordinary differential equations is considered in the context of global error estimation by using adjoint-equation based methods. In the case of solvers based on the fixed leading coefficient backward differentiation formulae, the quality of defect estimates is shown to play a major role in the reliability of the global error estimator of Cao and Petzold. New defect estimates obtained by sampling the defect are derived to improve the quality and efficiency of adjoint-based global error estimation. The inclusion of only one estimate of the defect per timestep is shown to provide a good compromise between accuracy and efficiency for global error estimation of odes and method-of-lines pdes.

Key words. ODEs, Global Error Estimation, Adjoint Methods, Defect, Residual Error

AMS subject classifications.

1. Introduction.

The importance of obtaining reliable error estimation for solutions to time-dependent ordinary differential equations (odes) and partial differential equations (pdes) is well understood, see [4, 5, 13, 15, 16, 17, 18]. As mentioned in Cao and Petzold [4], many methods of global error estimation have been proposed, studied carefully and implemented in several ode solvers. These error estimators either use residual errors for the error indicators or use error recovery techniques. Residual errors are the errors that result when the numerical solutions fail to satisfy the differential equations exactly everywhere. The global error estimates that use error recovery techniques often solve the problem a second time with a reduced step size or tolerance and assume the second integration is more accurate, the error in the first integration is then recovered by the differences between the two numerical solutions. These estimates may sometimes be inaccurate since the second integration may not yield a more accurate solution [4], [22]. As a consequence, there have been many error estimates that use residual errors and a multiplier based on the solution of an adjoint equation as a means of estimating the global error, [4].

The term "residual error" is used in Cao and Petzold [4] to mean "defect" as proposed and used by many authors; see, for example, [7, 9, 12, 21, 24, 25]. Calculated estimates of the defect are also sometimes used to gain confidence in a numerical solution. In order to control global error in numerical solutions of odes, many control strategies either control the local error or the defect, see [7, 8, 9, 13, 15, 21, 22, 23]. In some sense, defect control is more powerful than local error control when defect indicates the error throughout the interval of integration and local error indicates the error only in advancing a step [22]. As a consequence, controlling of defect is mentioned as being an appropriate method for calculating a more reliable numerical solution. More generally two important classes of global error estimates for initial value ordinary differential equations are the classical approach (based on forward integration of an error equation) and the adjoint approach that uses defect as an error indicator [4, 14] and muliplies it by the solution to an adjoint problem to estimate the

[*] This work was supported by the U.S. Department of Energy through the Center for the Simulation of Accidental Fires and Explosions (C-SAFE) under grant W-7405-ENG-48.

[†] (ltran@cs.utah.edu).

[‡] (mb@cs.utah.edu).

error. These approaches are compared by [14] for their reliability and efficiency. One disadvantage of adjoint-based methods is the need to store the forward solution that is required during the backwards time integration. Lang and Verwer [14] suggested that the adjoint method may not be competitive against the classical approach due to its huge storage demand for large problems, even though both approaches work well in terms of reliability. On the other hand, Cao and Petzold [4] suggest that the adjoint method is an attractive choice for estimating the error, and also suggested a novel approach based on the small sample method for reducing the number of backward time integrations used by the adjoint method. Furthermore the adjoint systems are linear and can be computed in parallel. Even though solving for the adjoint system requires extra work and storage, the adjoint solutions are useful to adaptively control global error as they are the appropriate weighting coefficients of local errors in forming the estimate of the global error.

In this paper, we will show how to improve the adjoint-based global error estimate proposed by Cao and Petzold [4] by sampling the defect and by removing an assumption used by those authors. The starting point for this work is a description of the Cao and Petzold method and some numerical results illustrating its performance. While defect estimates are available for many time integration methods; see, for example, [12, 14], the lack of a reliable defect estimate for variable order and variable step BDF methods is a challenge for global error estimation using the adjoint method. As BDF methods are widely used for obtaining solutions for stiff differential equations and differential algebraic equations, we shall propose a new defect estimate based on defect sampling. Defect estimates for Adams PECE codes are described in detail by Higham [12] and provide a useful starting point for the estimate of the BDF defect described here in Section 3. One approach to the estimation of the complex form of the defect estimate for variable step/variable order BDF methods seems to require sampling at several internal points, depending on the order of the method, Although this approach is successful, it is not cost-effective, as is shown in Section 4. In Section 5 of this paper we will show that it is possible to derive defect error estimates for variable order and variable step BDF methods that are both sufficiently reliable and less expensive in terms of the number of sample points needed. The use of only one sample point per timestep provides a good compromise between accuracy and efficiency, as shown by the results computed using the IDA solver based on DASSL.

In the method of lines context this paper extends the forward integration approach of [1] by using the adjoint-based method to estimate the combined spatial and temporal error. The main advantage of this method is that the adjoint systems for both spatial and temporal error are identical. This makes it possible to estimate the combined spatial and temporal error, and, opens the possibility for spatial and temporal error control.

2. Adjoint error estimation for odes.

The starting point for this work is the adjoint-based global error estimate of Cao and Petzold [4] which will be described here in a slightly modified form from that given in their paper. The class of odes considered here is given by:

$$\begin{cases} \dot{x}(t) & = f(x,t), \quad 0 \le t \le T, \\ x(0) & = x_0 \end{cases} \tag{2.1}$$

where $x \in R^n$. The numerical solution $\tilde{x} \in R^n$ satisfies the following perturbed system, [4]:

$$\begin{cases} \dot{\tilde{x}}(t) &= f(\tilde{x},t) + r(t), \quad 0 \le t \le T, \\ x(0) &= x_0 + r_0 \end{cases} \tag{2.2}$$

where $r(t)$ is the pertubation of the numerical solution at time t and the initial pertubation r_0 is at time $t = 0$. Hereafter, the term "defect" will be used to refer to $r(t)$.

Define $e(t) = \tilde{x}(t) - x(t)$ as the error in the numerical solution at time t. The system for the evolution of this error, [4] is then:

$$\begin{cases} \dot{e}(t) &= J(\tilde{x},t)e(t) + r_1(x,\tilde{x},t) + r(t) \\ e(0) &= r_0 \end{cases} \tag{2.3}$$

where $J(\tilde{x},t)$ is the Jacobian of f at $\tilde{x}$. The residual $r_1(x,\tilde{x},t)$ is an approximation to the quadratic and subsequent Taylor series terms given by $r_1(x,\tilde{x},t) = f(\tilde{x},t) - f(x,t) - J(\tilde{x},t)(\tilde{x} - x)$ with $\|r_1(x,\tilde{x},t)\|_\infty$ is assumed to be small when $\tilde{x}(t)$ is close to $x(t)$, as, if the original ode is not solved to sufficient accuracy, then the adjoint solution procedure and the global error estimate cannot be trusted, [4]. With the assumption that $\|r_1(x,\tilde{x},t)\|_\infty$ is small enough to be neglected,[4], we have:

$$\begin{cases} \dot{e}(t) &\approx J(\tilde{x},t)e(t) + r(t) \\ e(0) &= r_0. \end{cases} \tag{2.4}$$

Let $\lambda(t)$ be some vector in R^n that is the solution to the following system:

$$\begin{cases} \dot{\lambda}(t) &= -J^T(\tilde{x},t)\lambda(t), \quad 0 \le t \le T \\ \lambda(T) &= l \end{cases} \tag{2.5}$$

for some vector l in R^n. It is now possible to derive a simplified form of the error estimate procedure, [17]. Multiplying both sides of first equation in (2.4) by $\lambda^T(t)$ gives:

$$\begin{aligned} \lambda^T(t)\dot{e}(t) &\approx \lambda^T(t)J(\tilde{x},t)e(t) + \lambda^T(t)r(t) \\ &= (J^T(\tilde{x},t)\lambda(t))^T e(t) + \lambda^T(t)r(t) \\ &= (-\dot{\lambda}(t))^T e(t) + \lambda^T(t)r(t). \end{aligned}$$

Rearranging this yields:

$$\lambda^T(t)\frac{d}{dt}e(t) + \frac{d}{dt}(\lambda^T(t))e(t) \approx \lambda^T(t)r(t), \tag{2.6}$$

and, in turn, gives:

$$\frac{d}{dt}(\lambda^T(t)e(t)) \approx \lambda^T(t)r(t). \tag{2.7}$$

Integrating both sides of the above equation gives:

$$\begin{aligned} \int_0^T \frac{d}{dt}(\lambda^T(t)e(t))dt &\approx \int_0^T \lambda^T(t)r(t)dt, \\ \lambda^T(T)e(T) - \lambda^T(0)e(0) &\approx \int_0^T \lambda^T(t)r(t)dt, \\ l^T e(T) - \lambda^T(0)r_0 &\approx \int_0^T \lambda^T(t)r(t)dt. \end{aligned}$$

Or:

$$l^T e(T) \approx \int_0^T \lambda^T(t) r(t) dt + \lambda^T(0) r_0. \tag{2.8}$$

It is perhaps worth remarking that if we replace $r(t)$ by $r(t) + r_1(x, \tilde{x}, t)$ then this equation is exact. To estimate the ith-component of error vector e(T), we solve system (2.5) with initial condition $l = [0, 0, ..., 0, 1, 0, ...0]^T$ with a value of 1 at the ith-component and 0 elsewhere. So in order to estimate the global error vector e(T), we have to solve the system (2.5) n times, where n is the number of equations in ode system, with n different values of vector l: $l = e_1, e_2, ..., e_n$, where $e_1, e_2, ..., e_n$ are the unit vectors of R^n. Cao and Petzold [4] use the small sample statistical method to greatly reduce the number of these integrations. As the value of $\lambda(t)$ is only obtained numerically, the estimation of global error is valid only if the adjoint system in (2.5) is solved with sufficient accuracy.

An important part of the estimation of global error, as outlined above, is to estimate the defect $r(t)$, where $r(t)$ is defined by:

$$r(t) = \dot{\tilde{x}}(t) - f(\tilde{x}, t). \tag{2.9}$$

Besides obtaining the numerical solutions at the end of integration steps, the ode solver often provides an interpolation method to obtain the numerical solution at any point in between the steps. The defect r(t) is then available at any point in time.

2.1. Defect and global error estimation using the approach of Cao and Petzold. The estimates of the ode defect $r(t)$ and global error $e(t)$ of [4] are based upon the fixed leading coefficient backward differentiation formula used in the DASSL DAE Solver. DASSL uses divided formulae to represent the numerial solution to DAEs. Consider the case in which we have a k-th degree interpolating polynomial based on a set of nodal values $x(t_n)$,$x(t_{n-1})$,...,$x(t_{n-k})$ is defined using divided differences as defined by:

$$x[t_n, t_{n-1}, ..., t_{n-k}] = \frac{x[t_n, t_{n-1}, ..., t_{n-k+1}] - x[t_{n-1}, t_{n-2} ..., t_{n-k}]}{t_n - t_{n-k}}, \tag{2.10}$$

where $x[t_n] = x(t_n)$ and $x[t_n, t_{n-1}] = \frac{x[t_n] - x[t_{n-1}]}{t_n - t_{n-1}}$.

Suppose that a set of time levels are given by $t_n, t_{n-1}, t_{n-2}, ...$ with associated numerical solution values $\tilde{x}_n, \tilde{x}_{n-1}, \tilde{x}_{n-2}, ...$ then the standard Newton divided difference form of the interpolating polynomial used by DASSL is given by

$$\begin{aligned} x^p_{n+1}(t) = b_{0,n}(t)\ \tilde{x}[t_n] + b_{1,n}(t)\ \tilde{x}[t_n, t_{n-1}] + b_{2,n}(t)\ \tilde{x}[t_n, t_{n-1}, t_{n-2}] + ... \\ + b_{k,n}(t)\ \tilde{x}[t_n, t_{n-1}, t_{n-2}, ..., t_{n-k}], \end{aligned} \tag{2.11}$$

where

$$b_{0,n}(t) = 1, b_{1,n}(t) = (t - t_n), \quad b_{2,n}(t) = (t - t_n)(t - t_{n-1}), ... \quad . \tag{2.12}$$

Equation (2.11) is used to predict the numerical solution at any point in the interval $[t_{n-k}, t_{n+1}]$. The predicted derivative may be similarly written as

$$\begin{aligned} x^{'p}_{n+1}(t) = \frac{db_{1,n}(t)}{dt}\ \tilde{x}[t_n, t_{n-1}] + \frac{db_{2,n}(t)}{dt}\ \tilde{x}[t_n, t_{n-1}, t_{n-2}] + ... \\ + \frac{db_{k,n}(t)}{dt}\ \tilde{x}[t_n, t_{n-1}, t_{n-2}, ..., t_{n-k}]. \end{aligned} \tag{2.13}$$

BDF codes such as DASSL also make use of these polynomials to predict the numerical solution at the next time step. The system of equations solved for the new solution at time t_{n+1} is given by

$$x'^{p}_{n+1}(t_{n+1}) - \frac{\alpha_s}{h_{n+1}}\left(\tilde{x}_{n+1} - x^{p}_{n+1}(t_{n+1})\right) = f(t_{n+1}, \tilde{x}_{n+1}), \tag{2.14}$$

where $h_{n+1} = t_{n+1} - t_n$ and for a method of order k, $\alpha_s = -\sum_{i=1}^{k} \frac{1}{i}$. Substituting from equations (2.11-2.13) and multiplying by $\frac{h_{n+1}}{\alpha_s}$ enables this to be written in a more recognizable BDF form as

$$(\tilde{x}_{n+1} - \tilde{x}_n) - \sum_{j=1}^{k}\left[b_{j,n} + \frac{h_{n+1}}{\alpha_s}\frac{db_{j,n}}{dt}\right]\tilde{x}[t_{n-j}, ..., t_n] = \frac{h_{n+1}}{(-\alpha_s)} f(t_{n+1}, \tilde{x}_{n+1}). \tag{2.15}$$

This equation is used to solve for the numerical solution at t_{n+1}. The numerical solution at any point t that lies between t_n and t_{n+1} is obtained using the interpolating polynomial defined as in (2.11) but with a different set of nodal values $\tilde{x}_{n+1}, \tilde{x}_n, ..., \tilde{x}_{n+1-k}$. This polynomial may be rewritten in Lagrange form as:

$$\tilde{x}(t) = \sum_{i=0}^{k} \prod_{j=0, j\neq i}^{k} \frac{(t - t_{n+1-j})}{(t_{n+1-i} - t_{n+1-j})} \tilde{x}_{n+1-i}. \tag{2.16}$$

We use a shorthand notation $\amalg_i^{a..b}(t)$ for $\prod_{j=a,j\neq i}^{b} \frac{(t-t_{n+1-j})}{(t_{n+1-i}-t_{n+1-j})}$ for convenience of exposition. Using this shorthand notation in the Lagrange form of $\tilde{x}(t)$ gives:

$$\tilde{x}(t) = \sum_{i=0}^{k} \amalg_i^{0..k}(t)\tilde{x}_{n+1-i}. \tag{2.17}$$

A local solution on the interval $[t_n, t_{n+1}]$ as defined in [4] satisfies

$$\begin{cases} \dot{u}_{n+1}(t) = f(u_{n+1}(t), t) \qquad t \in [t_{n+1-k}, t_{n+1}], \\ u_{n+1}(t_{n+1-k}) = \tilde{x}_{n+1-k}. \end{cases} \tag{2.18}$$

Let $s_{n+1}(t)$ be the polynomial that interpolates $k+1$ points (t_{n+1}, s^0_{n+1}),(t_n, s^1_{n+1}), ... (t_{n-k+1}, s^k_{n+1}) where s^i_{n+1} is the notation for $s_{n+1}(t_{n+1-i})$ and $s^i_{n+1} = u_{n+1}(t_{n+1-i})$ for $i = 0..k$. The polynomial $s_{n+1}(t)$ is then written in Lagrange form as:

$$s_{n+1}(t) = \sum_{i=0}^{k} \amalg_i^{0..k}(t)s^i_{n+1}. \tag{2.19}$$

Then for any t in $[t_n, t_{n+1}]$, we have:

$$u_{n+1}(t) = s_{n+1}(t) + IE(t) \tag{2.20}$$

where IE(t) is interpolation error at t, and is defined as:

$$IE(t) = (t - t_{n+1})(t - t_n)...(t - t_{n-k+1})\frac{u^{(k+1)}_{n+1}(\tau)}{(k+1)!}, \tag{2.21}$$

for some value τ that lies in the interval $[t_{n-k+1}, t_{n+1}]$. Differentiating equation (2.17) gives:

$$\dot{\tilde{x}}(t) = \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t)\tilde{x}_{n+1-i} \tag{2.22}$$

for any t in $[t_n, t_{n+1}]$ and where $\dot{\Pi}_i^{a..b}$ is the time derivative of $\Pi_i^{a..b}$:

$$\dot{\Pi}_i^{a..b} = \sum_{j=a, j\neq i}^{b} \frac{1}{(t_{n+1-i} - t_{n+1-j})} \prod_{l=a, l\neq i,j}^{b} \frac{(t - t_{n+1-l})}{(t_{n+1-i} - t_{n+1-l})}. \tag{2.23}$$

Differentiating equation (2.20) gives:

$$\dot{u}_{n+1}(t) = \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t)s_{n+1}^i + I\dot{E}(t). \tag{2.24}$$

We now rewrite the defect, $r(t)$, for the perturbed system (2.2) on the interval $[t_n, t_{n+1}]$ using the definition of local solution in (2.18) as:

$$r(t) = \dot{\tilde{x}}(t) - f(\tilde{x}, t) = \dot{\tilde{x}}(t) - \dot{u}_{n+1}(t) + f(u_{n+1}(t), t) - f(\tilde{x}(t), t). \tag{2.25}$$

Cao and Petzold [4] assume that the function $f(x, t)$ is sufficiently smooth and satisfies the Lipschitz condition, $\|f(u_{n+1}(t), t) - f(\tilde{x}(t), t)\| \leq L\|u_{n+1}(t) - \tilde{x}(t)\|$ for some constant L. It is pointed out in [4] that if $|hL| \leq 1$ then $u_{n+1}(t) - \tilde{x}(t) = O(h_{n+1}^{k+1})$ while $\dot{u}_{n+1}(t) - \dot{\tilde{x}}(t) = O(h_{n+1}^k)$. Consequently the term $f(u_{n+1}(t), t) - f(\tilde{x}(t), t)$ may be disregarded as not making a significant contribution to the overall defect. This idea is also used by [6]. The defect is thus given by:

$$\begin{aligned} r(t) &\approx \dot{\tilde{x}}(t) - \dot{u}_{n+1}(t), \\ &\approx \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t)(\tilde{x}_{n+1-i} - s_{n+1}^i) - I\dot{E}(t). \end{aligned} \tag{2.26}$$

This equation may be rewritten as:

$$r(t) \approx \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t)d_{n+1}^i - I\dot{E}(t), \tag{2.27}$$

where $d_{n+1}^i = \tilde{x}_{n+1-i} - s_{n+1}^i$ for $i = 0..k$. From [4], $d_{n+1}^i = O(h_{n+1}^{k+1})$ and so:

$$r(t) \approx \sum_{i=0}^{k} h_{n+1}^{k+1}\dot{\Pi}_i^{0..k} C_0 - I\dot{E}(t), \tag{2.28}$$

where $C_0 = C_{k+1}x^{(k+1)}$ and

$$I\dot{E}(t) \approx \frac{x^{(k+1)}(\tau)}{(k+1)!} \sum_{i=0}^{k} \prod_{j=0, j\neq i}^{k} (t - t_{n+1-i}) \approx \sum_{i=0}^{k} \frac{1}{i+1} x^{(k+1)}(t)h_{n+1}^k. \tag{2.29}$$

The calculation of the above terms requires the estimation of $x^{(k+1)}(t)$ from the divided difference representation of the solution. The defect may then be written as:

$$r(t) \approx C(h_{n+1})h_{n+1}^{k} \tag{2.30}$$

where $C(h_{n+1})$ at step size h_{n+1} is calculated as in equations (24) and (26) of [4] as:

$$C(h_{n+1}) \approx C_{k+1}x^{(k+1)} \sum_{i=0}^{k} \dot{\amalg}_i^{0..k} h_{n+1} + \sum_{i=0}^{k} \frac{1}{(i+1)} x^{(k+1)}. \tag{2.31}$$

Taking $\lambda(t_n + \tau) \approx \bar{\lambda}(t_n)$ for $0 \leq \tau \leq h_{n+1}$, and letting $t_1 = 0, t_2, t_3, t_4, ..., t_{m+1} = T$ be the times at which the numerical solution is calculated, then the equation (2.8) is rewritten as:

$$l^T e(T) \approx \sum_{j=1}^{m} \bar{\lambda}^T(t_j) \int_{t_j}^{t_{j+1}} r(t)dt + \lambda^T(0)r_0. \tag{2.32}$$

Cao and Petzold [4] choose $\bar{\lambda}(t_j) = \lambda(t_j)$ and replace the defect r(t) in (2.32) with (2.30), to arrive at:

$$l^T e(T) \approx \sum_{j=1}^{m} \lambda^T(t_j) C(h_{j+1}) h_{j+1}^{k+1} + \lambda^T(0)r_0. \tag{2.33}$$

This equation with an appropriate chosen value of l is used by Cao and Petzold [4] to estimate the global error.

2.2. Numerical examples.

We have been using $e(T)$ to denote the global errror at time T. Let $\tilde{e}(T)$ be the estimation of this quantity, we define the error index, *eindex*, of the L2-Norm of these quantities as:

$$eindex = \frac{\|\tilde{e}(T)\|}{\|e(T)\|}. \tag{2.34}$$

In several sections of this paper, we will repeatedly use the following examples for testing purposes. Examples 1 to 4 are taken from [4] and Examples 5 and 6 are taken from [15]. Example 7 is used by Butcher to illustrate stiffness.
Example 1.

$$\begin{cases} \dot{x} & = \lambda x, \quad 0 < t \leq T, \\ x(0) & = x_0. \end{cases} \tag{2.35}$$

This problem is solved with three different cases:

$$\begin{aligned} \lambda \ &= 1, \ \ x_0 = 10^{-4}, T = 10.0, \\ \lambda \ &= -1, \ x_0 = 1.0, T = 1.0, \\ \lambda &= -20, x_0 = 1.0, T = 1.0. \end{aligned}$$

Example 2.

$$\begin{cases} \dot{x} & = -(0.25 + \sin \pi t)x^2, \quad 0 < t \leq 1.0, \\ x(0) & = 1.0. \end{cases} \tag{2.36}$$

The analytical solution is $x(t) = \pi/(\pi + 1 + 0.25\pi t - \cos\pi t)$.
Example 3.

$$\begin{cases} \dot{x_1} & = \frac{1}{2(1+t)}x_1 - 2tx_2, \quad 0 < t \leq 10.0, \\ \dot{x_2} & = \frac{1}{2(1+t)}x_2 + 2tx_1, \quad 0 < t \leq 10.0, \\ x(0) & = (1.0, 0.0). \end{cases} \tag{2.37}$$

The analytical solution is: $x(t) = (x_1(t), x_2(t)) = ((1+t)^{\frac{1}{2}}\cos(t^2), (1+t)^{\frac{1}{2}}\sin(t^2))$.
Example 4.

$$\begin{cases} \dot{x_1} & = -x_2, \\ \dot{x_2} & = -x_1, \quad 0 < t \leq 10.0, \\ x(0) & = (2 \times 10^{-4}, 0.0). \end{cases} \tag{2.38}$$

The analytical solution is: $x(t) = (x_1(t), x_2(t)) = (10^{-4}(e^t + e^{-t}), 10^{-4}(e^{-t} - e^t))$.
Example 5.

$$\begin{cases} \dot{x_1} & = x_2, \\ \dot{x_2} & = -x_1, \quad 0 < t \leq 50.0, \\ x(0) & = (0.0, 1.0). \end{cases} \tag{2.39}$$

The analytical solution is: $x(t) = (x_1(t), x_2(t)) = (\sin(t), \cos(t))$.
Example 6.

$$\begin{cases} \dot{x_1} & = x_1, \\ \dot{x_2} & = x_2 + x_1x_1, \\ \dot{x_3} & = x_3 + x_1x_2, \\ \dot{x_4} & = x_4 + x_1x_3 + x_2x_2, \\ \dot{x_5} & = x_5 + x_1x_4 + x_2x_3, \quad 0 < t \leq 1.0, \\ x(0) & = (1.0, 1.0, 0.5, 0.5, 0.25). \end{cases} \tag{2.40}$$

The analytical solution is: $x(t) = (x_1, x_2, x_3, x_4, x_5) = (e^t, e^{2t}, \frac{1}{2}e^{3t}, \frac{1}{2}e^{4t}, \frac{1}{2}e^{5t})$.
Example 7.

$$\begin{cases} \dot{x} & = -L(x - \sin\pi t)) + \pi cos(\pi t), \quad 0 < t \leq 1.0, \\ x(0) & = 0.0. \end{cases} \tag{2.41}$$

The analytical solution is $x(t) = sin(\pi t)$ and L is positive and may be large.

Applying the error estimator of (2.33) and comparing the estimate global errors against the exact global errors of the numerical solutions at the end of integration gives the error indices shown in Table 2.1. Cao and Petzold show in [4] that the global error may be controlled using their estimate of global error. Although their global error estimate gives a wide range of error indices as seen in Table 2.1, the global error is often overestimated. The assumptions in [4] that d^i_{n+1} are the same for $i = 0..k$, that the value of τ in interpolation error is identical to t_n, and that the step sizes for last k steps are constant are likely the cause of this. The fact that the error is sometimes overestimated helps ensure the good global error control results of [4], but also raises the issue of whether better global error estimates are possible. The first step in achieving this is to improve the defect estimates by the use of sampling.

Table 2.1
Error indices(eindex) of global error estimate using (2.33)

	TOL							
Example	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
$1(\lambda = 1)$	7.13	7.23	7.09	9.12	8.95	8.54	16.72	9.18
$1(\lambda = -1)$	13.41	3.22	2.96	129.2	6.74	2.45	8.67	5.73
$1(\lambda = -20)$	0.61	1.59	0.46	0.45	2.08	7.63	10.12	10.36
2	5.63	9.27	106.9	19.36	72.67	13.98	16.38	0.31
3	13.58	13.02	13.66	13.00	11.59	10.92	10.77	11.35
4	7.13	7.25	7.08	6.45	8.68	12.10	15.70	12.45
5	4.13	8.89	15.04	7.98	1.45	7.63	8.64	4.16
6	6.14	10.54	14.31	8.09	12.94	4.62	8.35	13.86
7	0.003	0.04	0.002	0.008	0.0001	0.0002	0.00007	0.00001

3. New Approach of global error estimation.

We start with the global error estimation equation in (2.8):

$$l^T e(T) \approx \int_0^T \lambda^T(t) r(t) dt + \lambda^T(0) r_0. \tag{3.1}$$

$$l^T e(T) \approx \sum_{j=1}^{m} \int_{t_j}^{t_{j+1}} \lambda^T(t) r(t) dt + \lambda^T(0) r_0. \tag{3.2}$$

Let rewrite the defect $r(t)$ as defined in (2.25):

$$r(t) = \dot{\tilde{x}}(t) - \dot{u}_{n+1}(t) - J(\tilde{x}(t), t)(\tilde{x}(t) - u_{n+1}(t)) \tag{3.3}$$

Then:

$$\begin{aligned}
\lambda^T(t) r(t) dt &= \lambda^T(t)(\dot{\tilde{x}}(t) - \dot{u}_{n+1}(t) - J(\tilde{x}(t), t)(\tilde{x}(t) - u_{n+1}(t))) \\
&= \lambda^T(t)(\dot{\tilde{x}}(t) - \dot{u}_{n+1}(t)) - \lambda^T(t) J(\tilde{x}(t), t)(\tilde{x}(t) - u_{n+1}(t))) \\
&= \lambda^T(t)(\dot{\tilde{x}}(t) - \dot{u}_{n+1}(t)) - (J^T(\tilde{x}(t), t)\lambda(t))^T(\tilde{x}(t) - u_{n+1}(t))) \\
&= \lambda^T(t)(\dot{\tilde{x}}(t) - \dot{u}_{n+1}(t)) + \dot{\lambda}^T(t)(\tilde{x}(t) - u_{n+1}(t))) \\
&= \frac{d}{dt}(\lambda^T(t)(\tilde{x}(t) - u_{n+1}(t))
\end{aligned}$$

Therefore:

$$\begin{aligned}
l^T e(T) &\approx \sum_{j=1}^{m} \int_{t_j}^{t_{j+1}} \frac{d}{dt}(\lambda^T(t)(\tilde{x}(t) - u_{n+1}(t)) dt + \lambda^T(0) r_0 \\
&\approx \sum_{j=1}^{m} \lambda^T(t_{j+1})(\tilde{x}(t_{j+1}) - u_{n+1}(t_{j+1})) - \lambda^T(t_j)(\tilde{x}(t_j) - u_{n+1}(t_j)) + \lambda^T(0) r_0 \\
&\approx \sum_{j=1}^{m} (\lambda^T(t_{j+1}) d^0_{j+1} - \lambda^T(t_j) d^1_{j+1}) + \lambda^T(0) r_0
\end{aligned}$$

4. Defect estimate using sampling. The idea of defect estimation and control is considered by many authors; see, for example, [4, 7, 9, 12, 14, 22]. These estimates and controls are for Runge-Kutta, Adams PECE codes, and variable order/variable step BDF. The idea of evaluating the defect at several points to form the defect over the span of a step is called defect sampling, [12], and will be used here to estimate the defect for BDF methods. In doing so we take into account the different values of d^i_{n+1} for different values of i, the value of τ in $x^{(k+1)}(\tau)$ not neccessarily being at the end of the integration step and non-uniform step sizes. Rewriting the formula for the defect as in (2.27) as:

$$r(t) \approx \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t) d^i_{n+1} - \dot{I}E(t)$$

where d^i_{n+1} is defined following equation (2.27) and defining

$$\pi(t) = \prod_{j=0}^{k} (t - t_{n+1-j}), \tag{4.1}$$

allows the defect estimate to be written as:

$$r(t) \approx \sum_{i=0}^{k} \dot{\Pi}_i^{0..k}(t) d^i_{n+1} - \dot{\pi}(t) \frac{u^{(k+1)}_{n+1}(\tau)}{(k+1)!}. \tag{4.2}$$

The quantities that constitute this estimate are d^i_{n+1} for $i = 0..k$ and $\frac{u^{(k+1)}(\tau)}{(k+1)!}$. The definition of local solution in (2.18) implies that $d^k_{n+1} = 0$. So at any step $[t_n, t_{n+1}]$ where the numerical solution at t_{n+1} is obtained via a BDF method of order k, we have to determine $k+1$ unknown quantities d^i_{n+1} for $i = 0..k-1$ and $\frac{u^{(k+1)}(\tau)}{(k+1)!}$ in order to estimate the defect on this step. This can be done by sampling the defect at $k+1$ points t^*_j for $j = 0..k$ in each step. When we substitute this analytical value of defect into the left hand-side of equation (4.2), we can construct one equation for the unknowns. So, for a scalar equation, when we sample the defect at $k+1$ points, we can construct a $(k+1) \times (k+1)$ linear system $Ax = b$ where:

$$A(i,j) = \begin{cases} \dot{\pi}(t^*_i) & if \quad j = k+1 \\ \dot{\Pi}_j^{0..k}(t^*_i) & otherwise \end{cases}$$

and:

$$b_i = r(t^*_i) \tag{4.3}$$

and the unknown vector $x^T = (d^0_{n+1}, d^1_{n+1}, ..., d^{k-1}_{n+1}, \frac{u^{(k+1)}(\tau)}{(k+1)!})$. For simplicity, the values of t^*_j are evenly spaced in the interval $[t_n, t_{n+1}]$. Solving the linear system $Ax = b$ for the unknown x provides a way of estimating the defect over the step. Once the values of x is determined, the estimate defect at any point in the interval $[t_n, t_{n+1}]$ is also known from (4.2). The estimation of the defect on this interval is the function obtained from (4.2) where $d^0_{n+1}, d^1_{n+1}, ..., d^{k-1}_{n+1}$ and $\frac{u^{(k+1)}(\tau)}{(k+1)!})$ are determined using sampling. In order to evaluate the performance of defect estimate, we compare the analytical value of defect $r(t)$ against the estimate defect $\tilde{r}(t)$. Define the index of defect estimation, *rindex*, as follows:

$$rindex(t) = \frac{\|\tilde{r}(t)\|}{\|r(t)\|}. \tag{4.4}$$

For each step, we evaluate the defect at 100 equidistant points and calculate the mean and the standard deviation (STD) for these values in the whole integration interval. Let $t_1 = 0, t_2, t_3, ..., t_{m+1} = T$ be the steps for time integration, the definitions for mean and STD are:

$$Mean = \frac{\sum_{i=1}^{m} \sum_{j=1}^{100} rindex(t_i + j * h_i/100)}{100 * m}, \tag{4.5}$$

$$STD = \sqrt{\frac{\sum_{i=1}^{m} \sum_{j=1}^{100} (rindex(t_i + j * h_i/100) - Mean)^2}{100 * m}}. \tag{4.6}$$

For a good defect estimate, the mean value is around 1.0 and the standard deviation is small. Looking at the results in Table 3.1, we see that the defect estimate using sampling is good for every example except the last.

Table 4.1

Mean and STD(standard deviation) of defect indices calculated from equations (4.5) and (4.6)

Example	TOL							
	10^{-3}		10^{-4}		10^{-5}		10^{-6}	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD
$1(\lambda = 1)$	1.0000	9.6e-11	1.0000	2.6e-10	1.0000	1.3e-8	1.0000	3.9e-8
$1(\lambda = -1)$	1.0000	2.04e-8	1.0000	6.8e-7	1.0000	0.0002	1.0000	0.0100
$1(\lambda = -20)$	1.0035	6.75e-9	1.0000	2.2e-6	1.0000	4.6e-5	1.0035	0.3704
2	1.0036	0.0709	1.0024	0.0839	1.0045	0.2469	1.0001	0.0025
3	1.0002	0.0026	1.0000	0.0012	0.9999	0.0008	1.0000	0.0005
4	1.0000	1.7e-10	1.0000	8.1e-10	1.0000	6.4e-9	1.0000	4.7e-8
5	1.0000	6.4e-10	1.0000	4.5e-8	1.0000	1.7e-5	1.0000	0.0012
6	1.0001	0.0019	1.0001	0.0019	1.0000	0.0006	1.0000	0.0015
7 $(L = 50)$	1.0268	0.3114	1.0184	0.2608	1.0138	0.2275	1.0100	0.1906

5. Adjoint-based global error estimation and local error. We use a different local problem from the one defined in (2.18) to derive an alternative method for calculating the defect. The modified local solution on the interval $[t_n, t_{n+1}]$ is denoted by $v_{n+1}(t)$ and is defined as the solution of

$$\begin{cases} \dot{v}_{n+1}(t) = f(v_{n+1}(t), t), & t \in [t_n, t_{n+1}], \\ v_{n+1}(t_n) = \tilde{x}_n. \end{cases} \tag{5.1}$$

The local error $le(t_{n+1})$ per step for the current step $[t_n, t_{n+1}]$ is then defined by

$$le(t_{n+1}) = v_{n+1}(t_{n+1}) - \tilde{x}_{n+1}. \tag{5.2}$$

Define, $g(t)$, a polynomial of degree k on $[t_{n+1-k}, t_{n+1}]$, that satisfies:

$$\begin{cases} g(t_{n+1-j}) = \tilde{x}_{n+1-j} & j = 1, ..., k \\ g(t_{n+1}) = v_{n+1}(t_{n+1}) \end{cases} \tag{5.3}$$

where $v_{n+1}(t_{n+1})$ is the solution to the system (5.1). It follows that g(t) is an interpolation polynomial of degree k that interpolates $k+1$ known points on the interval $[t_{n+1-k}, t_{n+1}]$ and approximates $v_{n+1}(t)$ on the interval $[t_n, t_{n+1}]$. The interpolation polynomial $g(t)$ may also be written in Lagrange form:

$$g(t) = \sum_{i=1}^{k} \amalg_i^{0..k}(t)\tilde{x}_{n+1-i} + \amalg_0^{0..k}(t)v_{n+1}(t_{n+1}). \tag{5.4}$$

Since g(t) approximates the local solution on the interval $[t_n, t_{n+1}]$, we have:

$$v_{n+1}(t) \approx g(t) + gt(t) \tag{5.5}$$

where the additional term gt may be written as a divided difference term:

$$gt(t) \approx \pi(t)[v_{n+1}(t_{n+1}), \tilde{x}_n, \tilde{x}_{n-1}, ..., \tilde{x}_{n-k}]. \tag{5.6}$$

and $\pi(t) = (t - t_{n+1})(t - t_n)...(t - t_{n+1-k})$. The defect, $r(t)$, may be then estimated as:

$$\begin{aligned} r(t) = \dot{\tilde{x}}(t) - f(\tilde{x}(t), t) &= \dot{\tilde{x}}(t) - \dot{v}_{n+1}(t) + f(v_{n+1}(t), t) - f(\tilde{x}(t), t) \\ &= \dot{\tilde{x}}(t) - \dot{v}_{n+1}(t) - J(\tilde{x}(t), t)(\tilde{x}(t) - v_{n+1}(t))). \end{aligned}$$

Using a similar derivation in section (3), we arrive at:

$$\lambda^T(t)r(t)dt = \frac{d}{dt}(\lambda^T(t)(\tilde{x}(t) - v_{n+1}(t)), \tag{5.7}$$

and then:

$$l^T e(T) \approx \sum_{j=1}^{m} -\lambda^T(t_{j+1})le(t_{j+1}) + \lambda^T(0)r_0. \tag{5.8}$$

6. Estimate error using defect sampling.

We have the defect as seen above:

$$r(t) \approx \dot{\tilde{x}}(t) - \dot{v}_{n+1}(t) - J(\tilde{x}(t), t)(\tilde{x}(t) - v_{n+1}(t)) \tag{6.1}$$

Assume that the jacobian $J(\tilde{x}(t), t)$ is close to $J(\tilde{x}(t_{n+1}), t_{n+1})$, multiply both sides of the above equation with $\lambda^T(t_{n+1})$ to get:

$$\begin{aligned} \lambda^T(t_{n+1})r(t) &= \lambda^T(t_{n+1})(\dot{\tilde{x}}(t) - \dot{v}_{n+1}(t)) + \dot{\lambda}^T(t_{n+1})(\tilde{x}(t) - v_{n+1}(t)) \\ &= \lambda^T(t_{n+1})(-\dot{\amalg}_0^{0..k}(t)le(t_{n+1}) - \dot{\pi}(t)ge(t_{n+1})) \\ &\quad + \dot{\lambda}^T(t_{n+1})(- \amalg_0^{0..k}(t)le(t_{n+1}) - \pi(t)ge(t_{n+1})) \\ &= \dot{\amalg}_0^{0..k}(t)(-\lambda^T(t_{n+1})le(t_{n+1})) + \dot{\pi}(t)(-\lambda^T(t_{n+1})ge(t_{n+1})) \\ &\quad + \amalg_0^{0..k}(t)(-\dot{\lambda}^T(t_{n+1})le(t_{n+1})) + \pi(t)(-\dot{\lambda}^T(t_{n+1})ge(t_{n+1})) \end{aligned}$$

So with 4 samples of the defect at $t1, t2, t3, t4$ we form the system:

$$\begin{bmatrix} \dot{\mathrm{II}}_0^{0..k}(t_1) & \dot{\pi}(t_1) & \mathrm{II}_0^{0..k}(t_1) & \pi(t_1) \\ \dot{\mathrm{II}}_0^{0..k}(t_2) & \dot{\pi}(t_2) & \mathrm{II}_0^{0..k}(t_2) & \pi(t_2) \\ \dot{\mathrm{II}}_0^{0..k}(t_3) & \dot{\pi}(t_3) & \mathrm{II}_0^{0..k}(t_3) & \pi(t_3) \\ \dot{\mathrm{II}}_0^{0..k}(t_4) & \dot{\pi}(t_4) & \mathrm{II}_0^{0..k}(t_4) & \pi(t_4) \end{bmatrix} \quad \begin{bmatrix} -\lambda^T(t_{n+1})le(t_{n+1}) \\ -\lambda^T(t_{n+1})ge(t_{n+1}) \\ -\dot{\lambda}^T(t_{n+1})le(t_{n+1}) \\ -\dot{\lambda}^T(t_{n+1})ge(t_{n+1}) \end{bmatrix} \quad \begin{bmatrix} \lambda^T(t_{n+1})r(t_1) \\ \lambda^T(t_{n+1})r(t_2) \\ \lambda^T(t_{n+1})r(t_3) \\ \lambda^T(t_{n+1})r(t_4) \end{bmatrix}$$

By solving this system, we may determine the value of $-\lambda^T(t_{n+1})le(t_{n+1})$ and use for error estimation in (5.8), thus giving an approximation to the defect:

$$\tilde{r}(t) = -\dot{\mathrm{II}}_0^{0..k} le(t_{n+1}) - \dot{g}t(t). \tag{6.2}$$

There are two quantities that determine the form of the estimated defect over a step: $le(t_{n+1})$ and $[v_{n+1}(t_{n+1}), \tilde{x}_n, \tilde{x}_{n-1}, ..., \tilde{x}_{n-k}]$. Using the same method as in Section 3, we may obtain the values of these quantities on each step by sampling the defect at 2 points in the interval $[t_n, t_{n+1}]$. In order to obtain the global error estimate in this case, we evaluate the term $\int_{t_n}^{t_{n+1}} \tilde{r}(t)dt$ as follows:

$$\begin{aligned} \int_{t_n}^{t_{n+1}} \tilde{r}(t)dt &= \int_{t_n}^{t_{n+1}} (\dot{\tilde{x}}(t) - \dot{v}_{n+1}(t)), \\ &= (\tilde{x}(t_{n+1}) - v_{n+1}(t_{n+1})) - (\tilde{x}(t_n) - v_{n+1}(t_n)), \\ &= -le(t_{n+1}), \end{aligned}$$

and arrive at the global error estimate:

$$l^T e(T) \approx \sum_{j=1}^{m} -\bar{\lambda}^T(t_j)le(t_j) + \lambda^T(0)r_0. \tag{6.3}$$

where the local error at any step, $le(t_j)$, is determined by sampling the defect at 2 points and constructing a 2×2 linear system using (6.2) to solve for the two unknown quantities in this equation. As we can see in Table 5.1 that the global error is estimated wth good accuracy using two defect samples per step.

Table 6.1
Error indices(eindex) for global error estimate using (6.3) with two defect samples per step

	TOL							
Example	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
$1(\lambda = 1)$	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.95
$1(\lambda = -1)$	1.00	0.96	0.97	0.99	0.96	0.96	0.96	0.98
$1(\lambda = -20)$	3.81	2.71	2.63	1.80	1.07	1.05	0.96	1.03
2	0.97	0.97	0.98	0.98	1.07	1.00	0.99	0.99
3	0.96	0.96	0.95	0.97	0.98	0.98	0.97	0.97
4	0.95	0.95	0.95	0.95	0.95	0.97	0.96	0.98
5	0.96	0.96	0.97	0.97	0.97	0.97	0.96	0.98
6	0.94	0.96	0.97	0.97	0.97	0.97	0.98	0.98
7 ($L = 50$)	2.69	2.19	1.87	1.80	1.46	1.03	1.04	1.05

7. Global error estimation with one defect sample per step. In order to reduce the cost still further, will make use of information from the BDF time integration to reduce the number of defect samples to one per time step. Let $x^p_{n+1}(t)$ be the predictor polynomial of equation (2.11) that interpolates the solution values at $k+1$ points $t_n, ..., t_{n-k}$ given by $\tilde{x}_{n+1-i}$ for $i = 1..k+1$. This polynomial $x^p_{n+1}(t)$ is written in Lagrange form as:

$$x^p_{n+1}(t) = \sum_{i=1}^{k+1} \Pi_i^{1..k+1}(t)\tilde{x}_{n+1-i}. \tag{7.1}$$

The local solution on the interval $[t_n, t_{n+1}]$ may be approximated by:

$$v_{n+1}(t) \approx x^p_{n+1}(t) + (t-t_n)(t-t_{n-1})...(t-t_{n-k})[v_{n+1}(t_{n+1}), \tilde{x}_n, ..., \tilde{x}_{n-k}]. \tag{7.2}$$

Let $gtt(t) = (t-t_n)(t-t_{n-1})...(t-t_{n-k})[v_{n+1}(t_{n+1}), \tilde{x}_n, ..., \tilde{x}_{n-k}]$, then:

$$v_{n+1}(t) \approx x^p_{n+1}(t) + gtt(t). \tag{7.3}$$

The residual on the interval $[t_n, t_{n+1}]$ may be approximated by:

$$r(t) \approx \dot{\tilde{x}}(t) - \dot{v}_{n+1}(t) - J(\tilde{x}(t), t)(\tilde{x}(t) - v_{n+1}(t)). \tag{7.4}$$

Using a similar derivation to previous section, we have:

$$\lambda^T(t)r(t) \approx \frac{d}{dt}(\lambda^T(t)[\tilde{x}(t) - v_{n+1}(t)]) \tag{7.5}$$

The equation for the global error estimate is then given by:

$$\begin{aligned} l^T e(T) &\approx \sum_{j=1}^{m} \int_{t_j}^{t_{j+1}} \lambda^T(t)r(t)dt \\ &\approx \sum_{j=1}^{m} \lambda^T(t_{j+1})[\tilde{x}(t_{j+1}) - v_{n+1}(t_{j+1})] - \lambda^T(t_j)[\tilde{x}(t_j) - v_{n+1}(t_j)] \\ &\approx \lambda^T(t_{j+1})[\tilde{x}(t_{j+1}) - x^P_{n+1}(t_{j+1}) - gtt(t_{j+1})]. \end{aligned}$$

Therefore:

$$l^T e(T) \approx \lambda^T(t_{j+1})[\tilde{x}(t_{j+1}) - x^P_{n+1}(t_{j+1}) - gtt(t_{j+1})]. \tag{7.6}$$

As the values of $\tilde{x}_{n+1}$ and x^p_{n+1} are available at each step, we need only to estimate $gtt(t_{n+1})$ in order to use the above equation for global estimation. As the unknown quantity that we have to estimate is $[v_{n+1}(t_{n+1}), \tilde{x}_n, ..., \tilde{x}_{n-k}]$. Using the method of sampling, we could use the equation (7.4) to estimate this quantity. However, to simplify the calculation we drop the Jacobian term in that equation and arrive at:

$$\begin{aligned} r(t) &\approx \dot{\tilde{x}}(t) - v_{n+1}(t) \\ &\approx \dot{\tilde{x}}(t) - \dot{x}^p_{n+1}(t) - gtt(t) \\ &\approx \dot{\Pi}_0^{0..k}(t)(\tilde{x}(t_{n+1}) - x^p_{n+1}(t_{n+1})) - \pi(t)[v_{n+1}(t_{n+1}), \tilde{x}_n, ..., \tilde{x}_{n-k}]. \end{aligned}$$

With only one sample of the defect, we are able to estimate $[v_{n+1}(t_{n+1}), \tilde{x}_n, ..., \tilde{x}_{n-k}]$. The error indices for global error estimation using (7.6) are shown in Table 5.2. Although the global error estimate using one defect sample is not as good as the methods with two or more samples, it a considerable improvement on Table 2.1 and a good compromise between accuracy and efficiency.

TABLE 7.1
Error indices of global error estimate using (7.6) with one defect sample per step

	TOL							
Example	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
$1(\lambda = 1)$	0.47	0.59	0.85	0.93	1.02	1.06	1.08	1.13
$1(\lambda = -1)$	1.28	1.12	1.13	1.08	1.22	0.95	1.12	1.19
$1(\lambda = -20)$	1.16	1.06	1.06	1.46	1.35	1.35	0.96	1.32
2	1.23	1.12	1.38	1.24	0.61	1.22	1.17	1.18
3	1.26	1.19	1.16	1.18	1.17	1.17	1.17	1.16
4	0.47	0.61	0.85	0.95	1.01	1.09	1.11	1.14
5	1.21	1.18	1.19	1.17	1.17	1.18	1.15	1.17
6	1.01	1.09	1.12	1.15	1.15	1.18	1.17	1.18
$7\ (L = 50)$	0.85	1.03	0.96	0.80	2.91	1.45	1.41	1.45

8. Method of lines pde solution. In order to consider the application of the above techniques to the method of lines solution of pde problems consider the class of equations given by:

$$u_t = F(t, u, u_x, u_{xx}) \tag{8.1}$$

where $(x, t) \in \Omega = [a, b] \times (0, t_e]$. The boundary conditions are taken to be of the form:

$$u_x|_{x=a}(x, t) = g_a(x, t), \quad \forall t \in (0, t_e] \tag{8.2}$$

$$u_x|_{x=b}(x, t) = g_b(x, t), \quad \forall t \in (0, t_e]. \tag{8.3}$$

The initial condition has the form:

$$u(x, 0) = k(x), \quad \forall x \in [a, b]. \tag{8.4}$$

Consider a space discretization grid Ω_H with H is the length of discretization resulting in the space discrete points: $x_0, x_1, x_2, ..., x_N$. The solution $u_H(t)$ computed at these discrete points using time integration is given by:

$$U_H(t) = \left[U_H(x_0, t), U_H(x_1, t_1), U_H(x_2, t_1), ..., U_H(x_N, t_1)\right]^T, \tag{8.5}$$

where $U_H(x_i, t)$ is the solution to the p.d.e at mesh point x_i at time t. With the space discretization grid Ω_H , using finite differences for approximations of partial derivatives and treating the boundary conditions for the pde (see [1]), we arrive at the system of differential equations in time:

$$\begin{cases} \dot{U}_H(t) & = F_H(t, U_H(t)) \\ U_H(0) & = U_{0H}, \end{cases} \tag{8.6}$$

where the initial condition U_{0H} is determined by the initial condition of the pde. The approximation of $U_H(t)$ is $\tilde{U}_H(t)$ and is computed via time integration method.

9. Pde spatial and temporal error estimation. Define the temporal error for the given pde system by $et_H(t)$ where

$$et_H(t) = U_H(t) - \tilde{U}_H(t) \tag{9.1}$$

and where $\tilde{U}_H(t)$ is the perturbed solution due to numerical error from time integration. From Section 2, $et_H(t)$ satisfies the system:

$$\begin{cases} \dot{et}_H & = J_H(\tilde{U}_H, t)et_H + r_H(t) \\ et_H(0) & = r_{0_H} \end{cases} \tag{9.2}$$

where $J_H(\tilde{U}_H, t)$ is Jacobian of F_H at $\tilde{U}_H$ and $r_H(t) = \dot{\tilde{U}}(t) - F_H(t, \tilde{U}_H(t))$. While performing the space discretization, we also introduce error at the mesh points called space discretization error: $es_H(t) = u_H(t) - U_H(t)$. Where $u_H(t)$ is the restriction of the pde exact solution to the mesh. From [1], the equation for space discretization error is:

$$\begin{cases} \dot{es}_H & = J_H(\tilde{U}_H, t)es_H + TE_H(t) \\ es_H(0) & = 0 \end{cases} \tag{9.3}$$

where $TE_H(t) = \dot{u}_H(t) - F_H(u_H(t), t)$. So the overall error at spatial mesh points for space discretization grid Ω_H at any time t is $E_H(t) = es_H(t) + et_H(t)$ and $E_H(t) = u_H(t) - \tilde{U}_H(t)$. From equations (7.1) and (7.3), we have:

$$\begin{cases} \dot{E}_H & = J_H(\tilde{U}_H, t)E_H + r_H(t) + TE_H(t) \\ E_H(0) & = r_{0_H}. \end{cases} \tag{9.4}$$

We perform a similar derivation as in Section 2 by considering the adjoint ode system:

$$\begin{cases} \dot{\lambda}(t) & = -J_H^T(\tilde{U}_H, t)\lambda(t), \quad 0 \le t \le T \\ \lambda(T) & = l \end{cases} \tag{9.5}$$

for some vector l in R^n. Given the similar form of equations (7.2) and (7.3), we arrive at the spatial error estimate for time-dependent pdes from using the adjoint method:

$$l^T es_H(T) = \int_0^T \lambda^T(s) TE_H(s) ds + O(\delta^2). \tag{9.6}$$

The combination of spatial and temporal error for time-dependent pdes is:

$$l^T E_H(T) = \int_0^T \lambda^T(s)(r_H(s) + TE_H(s))ds + \lambda^T(0)r_{0_H} + O(\delta^2). \tag{9.7}$$

Exactly as in Section 2, once the vector l is chosen, the value in each component of error vector E_H at end time T is estimated using solution to the adjoint system (7.5) with ode defect and pde truncation error. With the assumption that $\lambda(t_n + \tau) \approx \lambda(t_n)$ for $0 \le \tau \le h_{n+1}$. The pde spatial and the combinated spatial and temporal errors are then estimated by:

$$l^T es_H(T) = \sum_{j=1}^{m} \lambda^T(t_j) \int_{t_j}^{t_{j+1}} TE_H(t)dt, \tag{9.8}$$

$$l^T E_H(T) = \sum_{j=1}^{m} \lambda^T(t_j) \int_{t_j}^{t_{j+1}} (r_H(t) + TE_H(t))dt + \lambda^T(0) r_{0_H} + O(\delta^2). \tag{9.9}$$

With $le(t_j)$ defined as in (5.2) and estimated using 2 defect samples, we have:

$$l^T E_H(T) \approx \sum_{j=1}^{m} \lambda^T(t_j)(le(t_j) + (t_{j+1} - t_j) TE_H(t_j)) + \lambda^T(0) r_{0_H} \tag{9.10}$$

9.1. Pde truncation error estimation using Richardson extrapolation and mesh refinement. Consider a fine-grid Ω_h where $h = \frac{H}{2}$. Let Ω_h be the actual mesh used to compute the numerical solution to the PDE and also be the ”fine” mesh in the Richardson extrapolation. Then let Ω_H be the ”coarse” mesh. The following results are from [1]. Let

$$TE_h(t) = \dot{u}_h(t) - F_h(u_h(t), t). \tag{9.11}$$

Then

$$TE_H(t) = \frac{4}{3}[\dot{U^h}_H(t) - F_H(t, U_H^h(t))] + \frac{4}{3}[\dot{et}_H - \frac{\partial F_H}{\partial u_H(t)} et_H] \tag{9.12}$$

$$[TE_h(t)]_{2i-1} = \frac{1}{4}[TE_H(t)]_i + \bigcirc(h^3) \tag{9.13}$$

$$[TE_h(t)]_{2i} = \frac{1}{8}([TE_H(t]_i + [TE_H(t)]_{i+1}) \tag{9.14}$$

Where $\dot{U}_H^h(t)$ and $U_H^h(t)$ are the restriction of solutions from the fine mesh to the coarse mesh.

10. Numerical results. For following problems, we solve the pdes on the spatial domain $[A, B]$ with time interval $(0, T]$. Let NPTS be the number of points that we use to perform spatial discretization. The mesh size is then $H = \frac{(B-A)}{(NPTS-1)}$. So for mesh Ω_H and calculated solution $\dot{\tilde{U}}_H$ and $\tilde{U}_H$ on this mesh, we construct a ”coarse” mesh with mesh size $K = 2H$, and then calculate truncation error on this ”coarse” mesh by:

$$TE_K(t) = \frac{4}{3}[\dot{\tilde{U}}_H(t) - F_K(t, \tilde{U}_H(t))] \tag{10.1}$$

The truncation errors on original mesh are then recovered using [1]:

$$[TE_H(t)]_{2i-1} = \frac{1}{4}[TE_K(t)]_i \tag{10.2}$$

$$[TE_H(t)]_{2i} = \frac{1}{8}([TE_K(t]_i + [TE_K(t)]_{i+1}) \tag{10.3}$$

The residual error $r_H(t)$ is approximated by following the method described in Section 5. By solving the ode systems of pdes using different local tolerances(TOL), we may estimate spatial and overall errors to the numerical solutions to pdes using equation

(7.8) and (7.9). With $E_H(T)$ be the vector of total discrete error for the pde at time T and $\tilde{E}_H(T)$ be the approximation of total error at time T. We define the overall error 'index' as follows:

$$index = \frac{\|\tilde{E}_H(T)\|}{\|E_H(T)\|}. \tag{10.4}$$

The following test examples will be used to investigate the performance of the error estimator defined by equation (7.10).

10.1. Problem 1. Problem 1 is the heat equation with Neumann boundary conditions:

$$\frac{\delta u}{\delta t} = \frac{\delta^2 u}{\delta x^2} \tag{10.5}$$

where $(x, t) \in [0, 1] \times (0, 0.2]$. With the boundary conditions:

$$\frac{\delta u}{\delta x}(x, t) = \pi e^{-\pi^2 t} \cos(\pi x) \tag{10.6}$$

at x=0 and x=1. The initial condition is consistent with the analytic solution:

$$u(x, t) = \sin(\pi x) e^{-\pi^2 t}. \tag{10.7}$$

TABLE 10.1
Error indices index for Example 1 using (7.10)

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	1.01	1.01	0.98	0.97	0.97	0.97	0.97	0.97
21	1.01	1.01	1.00	0.99	1.00	1.00	1.00	0.99
41	0.99	1.01	1.01	1.01	1.01	1.01	1.01	1.01
81	0.99	0.99	1.01	1.01	1.01	1.01	1.01	1.01
161	0.97	0.98	0.99	0.99	1.02	1.03	1.02	1.02
321	0.97	0.97	0.98	0.98	1.02	1.04	1.03	1.02

10.2. Problem 2. Problem 2 is an example of a problem with a non-linear source term and a travelling wave solution:

$$\frac{\delta u}{\delta t} = \frac{\delta^2 x}{\delta x^2} + u^2(1 - u) \quad (x, t) \in (0, 10) \times (0, 1.0] \tag{10.8}$$

with Dirichlet boundary conditions and initial conditions consistent with the analytic solution of

$$u(x, t) = \frac{1}{1 + e^{p*(x - pt)}} \tag{10.9}$$

where $p = 0.5\sqrt{2}$.

Table 10.2
Error indices index for Example 2 using (7.10)

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	0.92	0.90	0.89	0.88	0.88	0.87	0.87	0.87
21	0.98	0.98	0.97	0.95	0.97	0.97	0.97	0.97
41	1.01	1.01	1.01	0.96	0.99	0.99	0.99	0.99
81	1.02	0.99	1.00	0.89	1.00	1.00	1.01	1.01
161	1.67	0.98	0.99	0.92	1.01	1.01	1.01	1.01
321	2.03	0.97	0.98	0.98	1.02	1.01	1.01	1.01

10.3. Problem 3. This problem has a nonlinear source term and with nonlinear boundary conditions:

$$\frac{\delta u}{\delta t} = \frac{\delta^2 u}{\delta x^2} - 2\frac{\delta u}{\delta x}^2 \frac{1}{u} - (2+4t^3x)u^2 \tag{10.10}$$

where $(x,t) \in [0,1] \times (0,1]$ and boundary conditions:

$$\frac{\delta u}{\delta x}(0,t) = -u^2 \tag{10.11}$$

and

$$\frac{\delta u}{\delta x}(1,t) = -u^2(-2+t^4) \tag{10.12}$$

The initial conditions are consistent with analytic solution:

$$u(x,t) = \frac{1}{2-x^2+xt^4}. \tag{10.13}$$

Table 10.3
Error indices index for Example 3 using (7.10)

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	0.83	0.83	0.83	0.83	0.83	0.83	0.84	0.83
21	0.83	0.83	0.83	0.83	0.85	0.84	0.84	0.84
41	0.90	0.90	0.91	0.93	0.91	0.90	0.90	0.89
81	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96
161	1.05	1.02	1.02	1.02	1.01	1.01	1.01	1.00
321	2.50	1.28	1.08	1.08	1.04	1.03	1.02	1.02

11. Small sample statistical method. It has shown above that using an adjoint method is a good method for estimating error for discretized pdes. It is however insufficient since we have to solve adjoint system n times with n is the number of equations in the system. As mentioned in [4], if we allow the estimate to have a moderate relative error, the small sample statistical method might be used. The small sample statistical method was originally proposed and discussed in detail in [19], and was also described in [4]. The method generates two independent vectors l_1 and l_2 uniformly and randomly from the unit sphere S_{n-1} in n dimensions where n is the number of equations in the odes system and uses these vectors for l in equation (7.5). With these vectors, we obtain the values for $l_1^T e(T)$ and $l_2^T e(T)$ using (5.8). The expected values of $l_1^T e(T)$ and $l_2^T e(T)$ are given by:

$$E(|l_i^T e(T)|) = \|e(T)\| E_n, \quad i = 1, 2 \tag{11.1}$$

where $E_1 = 1, E_2 = 2/\pi$, and E_n can be estimated by $\frac{2}{\sqrt{\pi(n-1/2)}}$ for $n > 2$.

Following [4], we use $\xi_1 = \frac{|l_1^T e(T)|}{E_n}$ and $\xi_2 = \frac{|l_2^T e(T)|}{E_n}$ to estimate $\|e(T)\|$. For orthogonal random vectors l_1 and l_2, we have an estimate of $\|e(T)\|$ given by $\xi(2)$ where:

$$\xi(2) = E_2 \sqrt{\xi_1^2 + \xi_2^2}. \tag{11.2}$$

Let $c > 1$ be a given factor:

$$P(\frac{\|e(T)\|}{c} \leq \xi(2) \leq c\|e(T)\|) \approx 1 - \frac{\pi}{4c^2} \tag{11.3}$$

As is shown in [4], with two orthogonal random vectors yield an estimate of $\|e(T)\|$ which is correct to within a factor of 3 with 91% probability.

The results obtained by applying this approach to the examples in Section 8 gives the results recorded in Tables 9.1-9.3. With each example, we run the estimation of $\|e(T)\|$ with 2 orthogonal random vectors in S_{n-1} for 500 times and calculate the probability of $\|e(T)\|$ estimate that falls into the range $\frac{\|e(T)\|}{3}$ and $3\|e(T)\|)$ where $\|e(T)\|$ obtained in Section 8. The results in Table 9.1-9.3 show the consistency with the theory for the small sample statistical method.

Table 11.1
Probability of $\xi(2)$ that falls into $\frac{\|e(T)\|}{3}$ and $3\|e(T)\|)$ for Example 1

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	90.4%	90.2%	92.8%	91.2%	90.6%	91.4%	90.0%	91.2%
21	91.6%	93.8%	91.2%	90.0%	93.0%	91.2%	92.0%	93.2%
41	91.0%	91.6%	90.0%	94.8%	92.0%	91.8%	91.2%	90.6%
81	93.4%	92.8%	91.2%	92.2%	93.2%	93.6%	93.8%	92.4%
161	90.8%	93.0%	94.6%	93.4%	91.6%	91.4%	92.4%	92.2%
321	94.2%	92.4%	93.4%	91.2%	92.4%	91.2%	92.0%	93.6%

12. Control the error in a quantity of interest. It is often possible to formulate problems so that the quantity of interest for the user is posed as a coupled ode to the pde. This idea was goes back to the POST software of Schryer [20] and was used in the SPRINT1D software of Berzins et al. [2]. Given the above discussion of error estimation in odes and pdes, it is also straightforward to estimate global error of coupled system of pdes and odes. We will translate the pdes part into system of N equations of odes using the space discretization of Section 6 to obtain a system of odes with N+M equations where M is the number of equations for coupled odes. The residual errors for new system are derived as in Section 5, and truncation errors

Table 11.2
Probability of $\xi(2)$ *that falls into* $\frac{\|e(T)\|}{3}$ *and* $3\|e(T)\|$) *for Example 2*

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	89.2%	91.4%	94.2%	90.8%	91.8%	93.2%	92.2%	93.2%
21	91.4%	90.6%	91.6%	92.2%	93.8%	92.6%	91.2%	94.0%
41	93.2%	91.8%	92.0%	91.6%	92.2%	90.8%	91.4%	91.8%
81	92.8%	90.6%	90.6%	91.2%	90.4%	92.4%	91.6%	90.6%
161	93.4%	92.2%	91.8%	94.8%	90.6%	91.8%	90.0%	90.0%
321	92.0%	91.2%	90.8%	92.4%	90.6%	92.8%	93.2%	94.2%

Table 11.3
Probability of $\xi(2)$ *that falls into* $\frac{\|e(T)\|}{3}$ *and* $3\|e(T)\|$) *for Example 3*

	TOL							
$NPTS$	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
11	92.0%	94.0%	92.4%	92.2%	93.4%	91.6%	92.2%	93.2%
21	90.2%	93.2%	91.4%	91.2%	92.2%	90.6%	93.0%	93.4%
41	91.6%	90.2%	93.6%	93.4%	91.2%	92.6%	94.2%	90.2%
81	93.4%	91.0%	91.8%	91.8%	91.6%	92.4%	91.4%	92.2%
161	92.8%	92.4%	92.0%	92.6%	92.0%	93.4%	91.2%	91.4%
321	91.2%	90.4%	92.4%	91.6%	89.8%	91.2%	92.0%	93.4%

are derived as in Section 6 where truncation errors are zero for that part of the equations corresponding to coupled odes. The global error is then estimated using adjoint method as described as above.

12.1. Estimate the error in approximating the quantity of interest. In some cases, we are not interested in global error in every solution component, but rather in the error of a quantity of interest that involves a part of the solution of the pde system. If we could formulate the quantity of interest in the form of one or more extra odes, we are then able to solve for this pde/ode coupled system and obtain the quantity of interest. The adjoint method then may be used to estimate the error in approximating this quantity of interest. This idea may be demonstrated by the following examples:

12.1.1. Example 1. Consider pde system:

$$\begin{cases} u_t = u_{xx} + e^{-2u} + e^{-u}, (x,t) \in [0,1] \times (0,1.0] \\ u(x,0) = log(x+P) \\ u(0,t) = log(t+P) \\ u_x(1,t) = \frac{1}{x+t+P} \end{cases} \tag{12.1}$$

The analytical solution to this system is $u(x,t) = log(x+t+P)$. Suppose that we are interested in the total of $u(1,t)$ over time. The quantity of interest may be written as $\int_0^T u(1,t)dt$. If we let $\dot{v_1} = u(1,t)$, then: $\int_0^T u(1,t)dt = v_1(T) - v_1(0)$.

Combining the pde system with the ode for the quantity of interest, we have the coupled pde/ode system:

$$(12.2)\qquad \begin{cases} u_t = u_{xx} + e^{-2u} + e^{-u}, (x,t) \in [0,1] \times (0,1.0] \\ \dot{v_1} = u(1,t) \\ u(x,0) = log(x+P) \\ u(0,t) = log(t+P) \\ u_x(1,t) = \frac{1}{x+t+P} \end{cases}.$$

In order to estimate the error in the quantity of interest in this problem, we need to solve the adjoint system with condition $\lambda(T) = [0,0,0,...0,1]^T$. The result for approximating the error in estimating the quantity of interest is shown in Table 10.1. TOL is the local error tolerance used in time integration.

Table 12.1

Approximating error in estimation of quantity of interest described in 10.1 where Index is the fraction of approximate error over true error in estimated quantity of interest.

NPTS	True Error	Approximate Error	Index	TOL
20	-0.00252215	-0.00242941	0.96	1e-4
40	-0.00132927	-0.00123281	0.93	1e-4
80	-0.000668284	-0.000617518	0.92	1e-5
160	-0.000338754	-0.000313061	0.92	1e-5

12.1.2. Example 2. This pde comes from modeling the dual-sorption of percutaneous drug absorption and is defined as:

$$(12.3)\qquad \begin{cases} u_t = [1 + \frac{\alpha}{(1+\beta u)^2}]^{-1} u_{xx}, (x,t) \in (0,1) \times (0,T] \\ u(0,t) = 1 \\ u(1,t) = 0 \\ u(x,0) = 0. \end{cases}$$

The cumulative amount of drug eliminated into receptor cell per unit area at time T^* is $Ae^*(T^*)$ defined as:

$$(12.4)\qquad Ae^*(T^*) = -\delta \int_0^{T^*} u_x(1,t)dt.$$

Differentiating this equation gives:

$$(12.5)\qquad \dot{Ae^*} = -\delta u_x(1,t).$$

The equation for $\ddot{Ae^*}$ is then derived as:

(12.6)

$$\ddot{Ae^*} = -\delta u_{xt}(1,t) = -\delta u_{tx}(1,t) = -\delta \frac{d}{dx} u_t(1,t) = -\delta \frac{d}{dx} [1 + \frac{\alpha}{(1+\beta u)^2}]^{-1} u_{xx}(1,t).$$

After asigning $\alpha_1 = \frac{1}{1+\alpha}$ and $\alpha_2 = \frac{2\alpha\beta}{(1+\alpha)^2}$ we have:

$$(12.7)\qquad \ddot{Ae^*} = -\delta(\alpha_1 u_{xxx}(1,t) + \alpha_2 u_x(1,t) u_{xx}(1,t)).$$

We then arrive at the coupled system:

$$
\begin{cases}
u_t = [1 + \frac{\alpha}{(1+\beta u)^2}]^{-1} u_{xx}, (x,t) \in (0,1) \times (0,T] \\
u(0,t) = 1 \\
u(1,t) = 0 \\
u(x,0) = 0 \\
\dot{Ae}^* = -\delta u_x(1,t) \\
\ddot{Ae}^* = -\delta(\alpha_1 u_{xxx}(1,t) + \alpha_2 u_x(1,t) u_{xx}(1,t))
\end{cases}
\tag{12.8}
$$

Using the method of lines and approximating $u_x(1,t)$, $u_{xx}(1,t)$ and $u_{xxx}(1,t)$ with finite differences leads to a system of first-order ODEs. Using the DASSL time integration method for solving this system, we obtain the numerical solution of u(x,t) at grid points, and the values of Ae^*, and $\dot{Ae}^*$ at end time T. By choosing appropriate initial values for adjoint system, we can estimate the errors in approximating of Ae^* and $\dot{Ae}^*$. The lag-time at time T^* after vehicle removal is the value of t-intercept of the asymptote of the Ae^* at T^* versus time curve. So we have:

$$
t_{lag} = T^* - \frac{Ae^*(T^*)}{\dot{Ae}^*(T^*)}
\tag{12.9}
$$

and so the error in estimating t_{lag} may be approximated as:

$$
Error(t_{lag}) = \frac{Error(\dot{Ae}^*(T^*))Ae^*(T^*) - \dot{Ae}^*(T^*)Error(Ae^*(T^*))}{\dot{Ae^*}^2(T^*)}
\tag{12.10}
$$

Using the same parameters as in [10],$\alpha = 2.7995$ and $\beta = 2.709916$, $\delta = 19.36$, for donor concentration $C = 4.4mg/ml$, we can compare the computed error and truth error in approximating t_{lag}. The analytical value of t_{lag} is obtained by determining asymptotic steady-state solutions to Fick's second law for permeation. The following lag-times at t=12h after vehicle removal for different donor concentrations are obtained by using 19 inner points for space discretization and local tolerance for time integration is $1e-3$. The 'Error Index' is the quotient of 'Approximate Error' over 'True Error' as used above.

Table 12.2
Result for problem described in 10.2

C (mg/ml)	Exact Lag-time	Computed Lag-time	True Error	Approximate Error	Error Index
4.4	3.29	3.337	0.047	0.040	0.85
19.5	2.19	2.150	0.040	0.052	1.30
43.1	1.86	1.821	0.039	0.036	0.92
51.4	1.81	1.769	0.041	0.033	0.80
64.0	1.75	1.718	0.032	0.026	0.81

12.1.3. Example 3. A different example of an important quantity of interest arises from the modeling of the conditions under which a rod of explosive ignites. The governing differential equation here in [3]:

$$
u_t = 0.25 u_{xx} + 0.46 exp(33 - 1/u)
\tag{12.11}
$$

subject to the initial condition:

$$u(x,0) = u_1 \tag{12.12}$$

and boundary conditions:

$$\begin{aligned} u(0,t) &= u_0, \\ u(1,t) &= u_1, \\ u_0 &> u_1. \end{aligned}$$

The critical temperature of a rod solid explosive u_{CRIT} is the value at which $u_0 > u_{CRIT}$ leads to ignition whereas $u_0 < u_{CRIT}$ does not, where ignition is defined as $u(x,t) > u(0,t)$ at some position x that $0 < x < 1$ and at some time t that $t <= 10$.

In order to find the critical temperature, we follow the numerical technique used in [3]. For different spatial meshes, we obtain a different value of critical temperature, the initial temperature $u(0,t)$ such that the explosion occurs at $T = 10$, and different position u_i where $u_i(T) > u_0(T)$. However, there is an uncertainty in time of explosion due to the error in numerical value $\tilde{u}_i(t)$ where $\tilde{u}_i(t)$ is computed solution of $u_i(t)$.

Assume that we have determined the critical temperature u_{CRIT} and the computed value $\tilde{u}_i(T)$ with some global error $e_i(T)$, then we know the time of explosion must be $T + \Delta T$ and defined ΔT as uncertainty in time of explosion. To determine the time uncertainty ΔT, we use a rough calculation of u_i at $T + \Delta T$ as:

$$u_i(T + \Delta T) = u_i(T) + (u_i)_t(T)\Delta T. \tag{12.13}$$

The correct value $u_i(T)$ is approximated as: $u_i(T) = \tilde{u}_i(T) + e_i(T)$. Assume that $(u_i)_t$ is constant, we obtain the relationship:

$$\begin{aligned} \Delta T &= \frac{u_i(T+\Delta T) - u_i(T)}{(u_i)_t(T)} \\ &= \frac{u_i(T+\Delta T) - (\tilde{u}_i(T) + e_i(T))}{(u_i)_t(T)} \\ &\approx -\frac{e_i(T)}{(u_i)_t(T)} \end{aligned}$$

Table 12.3
Time uncertainty at T=10.0 with corresponding spatial mesh and critical temperature

H	0.015625	0.0078125	0.0039625
Critical Temperature	0.028379	0.028375	0.028372
$\tilde{u}_i$	0.0283797	0.0283751	0.0283721
Global Error of $\tilde{u}_i$	4.31e-05	1.01e-05	1.23e-06
$(u_i)_t$	1.91e-06	1.84e-07	3.93e-08
Time uncertainty(ΔT)	22.18	21.33	31.00

The calculation of time uncertainty for different spatial meshes is recorded in Table 10.3. Since the global error of $\tilde{u}_i(T)$ is significantly larger than that in $(u_i)_t(T)$, this results in a large value of time uncertainty. Table 10.3 also shows that a small change in u_0 might lead to a large change in the time of explosion. For example:

changing the value of u_0 from 0.028379 to 0.0283778 for the case $h = 0.015625$ causes the time of explosion to change from 10.0 to 32.0 and a change in value of u_0 from 0.028375 to 0.0283744 for the case $h = 0.0078125$ causes the time of explosion to change from 10.0 to 48.0. These results show that we have to take great care when calculating the resuts of the critical temperature problem.

13. Conclusions. The adjoit-based global error estimate introduced at the start of this paper has been improved by the use of defect sampling. Defect sampling is applied in each time interval to estimate the ode defect. The defect and thus the global error is best estimated with $k + 1$ defect samples per time step where k is the order of BDF method. However, the use of two samples per time step also yields a good estimation of global error. In order to reduce the cost, the number of defect samples is reduced to one by making use of information from IDA/DASSL. This is less accurate thabn the two sample approach but still useful for estimation of the global error in ode systems.

Spatial and temporal errors are the error sources associated with the discretization of time-dependent pdes when using the method of lines. The temporal error that correspond to the ode global error that may be estimated using adjoint-based method with defect sampling. Making use of the similarity between the systems of spatial and temporal error evolution, we can extend the adjoint method to spatial error estimation in which case the ode defect is replaced by pde truncation error. This pde truncation error may be estimated via Richardson extrapolation. Numerical results have shown that this approach works well for pde error estimation. Even though the adjoint method is somewhat inefficient in estimating the pde combined space and time error, the small sample statistical method has been shown to be able to estimate this error with a high probability.

The adjoint method for pde error estimation is also efficient when used with quantity of interest. Estimation of the error in the quantity of interest involves the estimation of some error component in solution vector of coupled pdes/odes. Finally this type of error estimation may also help in quantifying uncertainty in computed solution.

Acknowledgments. This work was supported by the U.S. Department of Energy through the Center for the Simulation of Accidental Fires and Explosions (C-SAFE) under grant W-7405-ENG-48.

REFERENCES

[1] M.Berzins, *Global error estimation in the method of lines for parabolic equations*, SIAM J. Sci. Statist. Comput., 9(4) (1988), pp. 687–703.

[2] M.Berzins, P.M.Dew and R.M.Furzeland *Developing Software for Time-Dependent Problems Using the Method of Lines and Differential Algebraic Integrators.* Applied Numerical Mathematics, 5, 375–397, 1989.

[3] G.B. Cook, *The initiation of explosions in solid secondary explosives*, Proc. Roy. Soc. London, A246 (1958), pp. 154–160.

[4] Y. Cao and L. Petzold, *A Posteriori Error Estimation and Global Error Control for Ordinary Differential Equations by the Adjoint Method*, SIAM J. Sci. Comput., 26(2) (2004), pp. 359–374.

[5] D.J. Estep, M.G. Larson, R.D. Williams, *Estimating the Error of Numerical Solutions of Systems of Reaction-Diffusion Equations*, Mem. Amer. Math. Soc, 696 (2000), pp. 1–109.

[6] W.H. Enright, *A New Error-Control for Initial Value Solvers*, Appl. Math. Comput., 31 (1989), pp. 588–599.

[7] W.H. Enright, *Continuous Numerical Methods for ODEs with Defect Control*, Journal of Computational and Applied Mathematics, 125(1-2) (2000), pp. 159–170.

[8] W.H. Enright and W.L. Seward, *Achieving Tolerance Proportionality in Software for Stiff Initial-Value Problems*, Journal of Computing, 42(1989), pp. 341–352.

[9] P.M. Hanson and W.H. Enright, *Controlling the Defect in Existing Variable-Order Adams Codes for Initial-Value Problems*, ACM Transactions on Mathematical Software, 9(1) (1983), pp. 71–97.

[10] A.B. Gumel, K. Kubota and E.H. Twizell, *A Sequential Algorithm for the non-linear dual-sorption Model of Percutaneous Drug Absorption*, Mathematical Biosciences, 152 (1998), pp. 87–103.

[11] D. Higham, *Global Error versus Tolerance for Explicit Runge-Kutta Methods*, IMA Journal of Numerical Analysis, 11(1991), pp. 457–480.

[12] D.J. Higham, *Defect Estimation in Adams PECE Codes*, Numerical Analysis Report No. 154, University of Manchester, England, 1988.

[13] D.J. Higham, *Global Error versus Tolerance for Explicit Runga-Kutta Methods*, IMA Journal of Numerical Analysis, 11(1991), pp. 457–480.

[14] J. Lang, J.G. Verwer, *On global error estimation and control for initial value problems*, SIAM J. Sci. Comput., 29 (2007),pp. 1460–1475.

[15] A. Logg, *Multi-Adaptive Error Control for ODEs*, Technical Report 98/20(1998), Oxford University, England.

[16] K.-S. Moon, A. Szepessy, R. Tempone and G.E. Zouraris , *Adaptive Approximation of Differential Equations Based on Global and Local Errors*, TRITA-NA-0006 (2000), NADA, KTH, Sweden.

[17] K.-S. Moon, A. Szepessy, R. Tempone and G.E. Zouraris , *A Variational Principle for Adaptive Approximation of ordinary Differential Equations.*, Numerische Mathematik, 93, 2003, 131-152.

[18] K.-S. Moon, A. Szepessy, R. Tempone and G.E. Zouraris , *Convergence Rates for Adaptive Approximation of ordinary Differential Equations.*, Numerische Mathematik, 93, 2003, 99-129.

[19] C. S. Kenney and A. J. Laub,*Small-sample statistical condition estimates for general matrix functions*, SIAM J. Sci. Comput., 15 (1994), pp. 36.61.

[20] N.L.Schryer, *POST - A Package for Solving Partial Differential Equations in One Space Variable*, Scientific Computing Group. Bell Labs, Lucent Technologies Report 84/4-1

[21] W.L. Seward, *Defect and Local Error Control in Codes for Solving Stiff Initital-Value Problems*, University of Toronto Technologies Report, 184/85 (1985)

[22] L.F. Shampine, *Error Estimation and control for ODEs*, Journal of Sci. Comput., 25(1) (2005),pp. 3–16.

[23] L.F. Shampine, *Solving ODEs and DDEs with Residual Control*, Appl. Numer. Math., 52 (2005), pp. 113–127.

[24] R.D. Skeel, *Global Error Estimation and the backward Differentiation Formulas*, Appl. Math. Comput., 31(1989),pp. 197–208.

[25] R.D. Skeel, *Thirteen ways to estimate global error*, Numer. Math. 48(1) (1986), pp. 1–20.