A Validation Method for Solution of Nonlinear Differential Equations: Construction of Exact Solutions Neighboring Approximate Solutions

Sangchul Lee*
Aerospace Engineering Center
Korea Aerospace Industries, Ltd., Sachon, Korea 664-942

Abstract

An inverse method is introduced to construct benchmark problems for the numerical solution of initial value problems. Benchmark problems constructed through this method have a known exact solution, even though analytical solutions are generally not obtainable. The solution is constructed such that it lies near a given approximate numerical solution, and therefore the special case solution can be generated in a versatile and physically meaningful fashion and can serve as a benchmark problem to validate approximate solution methods. A smooth interpolation of the approximate solution is forced to exactly satisfy the differential equation by analytically deriving a small forcing function to absorb all of the errors in the interpolated approximate solution. A multi-variable orthogonal function expansion method and computer symbol manipulation are successfully used for this process. Using this special case exact solution, it is possible to directly investigate the relationship between global errors of a candidate numerical solution process and the associated tuning parameters for a given code and a given problem. Under the assumption that the original differential equation is well-posed with respect to the small perturbations, we thereby obtain valuable information about the optimal choice of the tuning parameters and the achievable accuracy of the numerical solution. Illustrative examples show the utility of this method not only for the ordinary differential equations (ODEs) but for the partial differential equations (PDEs).

Key word : inverse method, benchmark problem, numerical solution, tuning parameters

Introduction

There are abundant numerical methods to obtain solutions of differential equations in dynamics and control area. Usually initial value problem is solved by various methods for simulation purpose [1], and two point boundary value problem in numerical optimization area can be solved by gradient method, shooting method, and method of particular solution [2]. In this paper, the initial value problems for nonlinear ODEs and hybrid ODE/PDE systems are considered. With most applications of approximate differential equation solution algorithms, we must somehow evaluate the accuracy of a given approximate solution, without knowing the true solution. What happens if we can construct an exact forced response solution for a special case motion near (in a sense to be established) a candidate approximate solution? This gives us an absolute standard and promises the capability of displaying exactly the space/time distribution of

* Principal Research Engineer
E-mail : tamu@koreaaero.com. TEL : 055-851-1828. FAX : 055-851-6977

solution errors for the special case solution and therefore suggesting remedies, if needed, to improve the discretization-based solution process. For this purpose, we introduce an inverse dynamics method for constructing exact special case solutions for hybrid ODE/PDE systems. In the robotics field, a similar idea, the "inverse dynamic formulation" or the "computed torque method" is described in [3,4]. If the desired motion is known, the inverse dynamic formulation algebraically solves the equations of motion to compute the forces and torques necessary to achieve that motion [3].

First, we restrict our concern to the initial value problem for nonlinear ODEs. In general, we do not know the true solution and any numerical method gives us an approximate solution; the numerical solutions generally contain two sources of error, round-off and truncation [5]. We must somehow evaluate the accuracy of a given approximate solution, typically without knowing the true solution. The most common way of assessing the true error of a numerical solution is to reduce some tolerance parameter, integrate again, and compare the results. Although more sophisticated error analyses can be conducted, there is no general way to absolutely guarantee the final accuracy of the solutions. This does not preclude obtaining practical solutions for most applications, but it remains very difficult to answer subtle questions.

In view of the historical and recent developments [5-7], we observe that the theory of differential equation solvers is far from complete, so that the understanding of a given code's performance invariably requires a study of experimental results. Hull, et al [8] and Krogh [9] provided two outstanding collections of test problems for this purpose. These test problems have been used in the development and testing of many codes and can be regarded as standard benchmark problems for initial value problem solvers. Whenever we know the true solutions of a test program, however, we can investigate the relationship between the true, or global error and the tuning parameters of a given code(e.g., step size, local error tolerance, order, etc.). The relationship between the behavior of an algorithm on a benchmark problem and the behavior of the algorithm on a problem of interest is difficult to establish. Since the problem of interest is almost never exactly solvable, we need a means to establish a customized benchmark problem which is a close neighbor of any given problem of interest. We introduce here a broadly applicable inverse method which constructs a neighbor of a given numerical approximate solution; the neighboring problem does in fact exactly satisfy the original differential equations(with a known, small forcing function) and serves as an excellent benchmark problem. More specifically, we present a broadly useful approach to construct a benchmark problem very near the problem of interest in a particular application. By virtue of the fact that the benchmark problem is a customized near neighbor of the problem of interest, we show that numerical convergence studies on the benchmark problem are directly useful in algorithm selection, tuning, and accuracy validation. Then, we generalize the idea to apply to hybrid ODE/PDE systems. The main difference is that there are two independent variables for space and time, thus we develop a two-variable orthogonal function expansion method.

In this paper, we propose a method to construct a benchmark problem which is a close neighbor of a given approximate solution of the original problem. The benchmark problem is constructed so that it satisfies exactly the differential equation but with a known, usually small, time varying forcing function. We can investigate the global error/parameter relationship of the benchmark problem with the true solution in hand. Under the assumption that the original problem is well-posed with respect to small perturbations, we have valuable information about the optimal parameters and the accuracy of the numerical solution. Actually the stability assumption is not so severe since any numerical method needs it more or less to obtain reliable solutions. Also, by introducing several neighboring approximate solutions with initial condition and parameter variations, then repeating the entire process, it is possible to experimentally establish insight on the size of the region over which the convergence properties are invariant. This methodology is useful in validating any given numerical solution method for both linear or nonlinear ODE and hybrid ODE/PDE system.

As an illustration, we demonstrate the idea using a simple nonstiff problem. We use the

Runge-Kutta fourth-order method with fixed step size. Therefore we have the most common case that the integration control parameter is simply the step size h. When we use the IMSL subroutines [10] DIVPRK and DIVPBS as solvers, we show the utility of this methodology for a celestial mechanics problem [9] that has been used as a test problem several times in the literature. Subroutine DIVPRK uses the Runge-Kutta formulas of order five and six developed by J. H. Verner. Subroutine DIVPBS uses the Bulirsh-Stoer extrapolation method and will terminate when impossible accuracies are specified. In the third example, we demonstrate the idea on an idealized three-body distributed-parameter system which has two independent variables for space and time. In the last example, we consider a typical stiff problem and discuss some limitations and restrictions of this methodology.

Construction of Exact Benchmark Problems

We want to construct new differential equations that are slightly perturbed versions of the original differential equations. For these new differential equations, we can establish the true analytical solution using an algebraic inverse idea. Then we can investigate the error/tolerance relationship with an absolute standard. Under local stability assumptions, we have valuable information about the optimal parameters and the accuracy of the particular numerical solution for the given original differential equations. The stability assumption is easily validated by constructing some neighboring benchmark problems.

Here we introduce one way for constructing exact benchmark problems. We take a global approach for the perturbation term instead of a piecewise polynomial perturbation to avoid the lack of smoothness at break points. First we consider the following two distinct initial value problems:

$$\begin{gathered} \dot{x} = f_1(x, t), \quad x(t_0) = x_0 \quad over \quad t_0 \le t \le t_f \\ f_1 : R^N \times R \rightarrow R^N \end{gathered} \tag{1}$$

$$\begin{gathered} \ddot{x} = f_2(x, \dot{x}, t), \quad x(t_0) = x_0, \quad \dot{x}(t_0) = \dot{x}_0 \quad over \ t_0 \le t \le t_f \\ f_2 : R^N \times R^N \times R \rightarrow R^N \end{gathered} \tag{2}$$

A candidate discrete approximate solution can be obtained from the original first and second order systems because these are certain drawbacks if one converts a naturally second order system into a first order system. To establish a continuous, differentiable motion near a given approximate solution, lease square approximation using the discrete version of the Chebyshev polynomials can be invoked to obtain the solution from the already discrete solution [11,12]. We first consider the least square approximation process. There are n data points denoted as

$$x_1 = g(t_1), \ x_2 = g(t_2), \cdot \cdot \cdot, x_n = g(t_n)$$

where t_i are the values of the equally spaced independent variable $(h_t = (t_{i+1} - t_i) = constant)$.

A linear transformation of independent variables should be made to use discrete orthogonality with weight function $w(t) = 1$,

$$\bar{t}(t) = \frac{t - t_1}{h_t}$$

where h_t is the constant increment of t,

$$x = g(t) - G(\bar{t}) \tag{3}$$

From n data points, the function G can be established as a linear combination of m basis functions that form the discrete version of the Chebyshev polynomials as follows:

$$G(\bar{t}) \equiv \sum_{i=1}^{m} a_i T_i(\bar{t})$$

where $m \le n$ and $T_i(\bar{t})$ is the ith Chebyshev polynomial.

The Chebyshev polynomials are defined as follows: If $u_m = m$ $(m=0,1,2,\cdots,N)$ and $w(u)=1$, then

$$T_n(u) = \sum_{m=0}^{n} (-1)^m \binom{n}{m}\binom{n+m}{m} \frac{u!(N-m)!}{(u-m)!N!}$$

With the recurrence relations:

$$T_0(u) = 1$$

$$T_1(u) = 1 - \frac{2u}{N}$$

$$(n+1)(N-n)T_{n+1}(u) = (2n+1)(N-2u)T_n(u) - n(N+n+1)T_{n-1}(u)$$

Note that the recurrence relations make it easy to evaluate an expansion in Chebyshev polynomials, and a similar recurrence makes it easy to evaluate the derivative of the expansion.

Using discrete orthogonality of the Chebyshev polynomials, the typical coefficient a_j can be obtained as follows:

$$a_j = \frac{\sum_{i=1}^{n} x_i T_j(\bar{t}_i)}{\sum_{i=1}^{n} T_j(\bar{t}_i) T_j(\bar{t}_i)}$$

where $1 \le j \le m$.

We can find $g(t)$ from $G(\bar{t})$ because $g(t) = G(\bar{t}(t))$. Using the least square approximation, we can find the continuous, differentiable, analytical solution $x(t)$ of Eq.(3) that interpolates the n discrete numerical solutions obtained from Eqs.(1) and (2). Now this analytical expression $x(t)$ does not satisfy exactly the Eqs.(1) and (2). However, substituting $x(t)$, $\dot{x}(t)$ into Eq.(1) allows us to determine an analytical function for the perturbation term $e_1(t)$ that appears in the following differential equation:

$$\dot{x}(t) = f_1(x(t), t) + e_1(t) \equiv F_1(x, t) \tag{4}$$

Alternatively, if the system is second order, then substituting $x(t)$, $\dot{x}(t)$, $\ddot{x}(t)$ into Eq.(2) allows us to determine the perturbation term $e_2(t)$ that appears in the following differential equation:

$$\ddot{x}(t) = f_2(x(t), \dot{x}(t), t) + e_2(t) \equiv F_2(x, \dot{x}, t) \tag{5}$$

Note that because $x(t)$, $\dot{x}(t)$, $\ddot{x}(t)$ are available functions, $F_1(x,t)$, $F_2(x, \dot{x}, t)$ are also available functions that satisfy Eqs.(4) and (5) exactly, and $x(t)$ is a neighbor of the original numerical solution $\{x_1, x_2, \cdots, x_n\}$. By construction, the functions $e_1(t) = \dot{x}(t) - f_1(x(t), t)$ and $e_2(t) = \ddot{x}(t) - f_2(x(t), \dot{x}(t), t)$ are known analytically and therefore these small forcing functions can be computed exactly at all t. These functions are programmed and Eqs.(4) and (5) can be solved by numerical methods and the results can be compared to the exact $x(t)$, $\dot{x}(t)$. The above mathematical procedure can be performed in an automated fashion using computer symbol manipulation [13]. The symbol manipulation can also automate the generation of C or FORTRAN code to compute function $e_1(t)$ and/or $e_2(t)$.

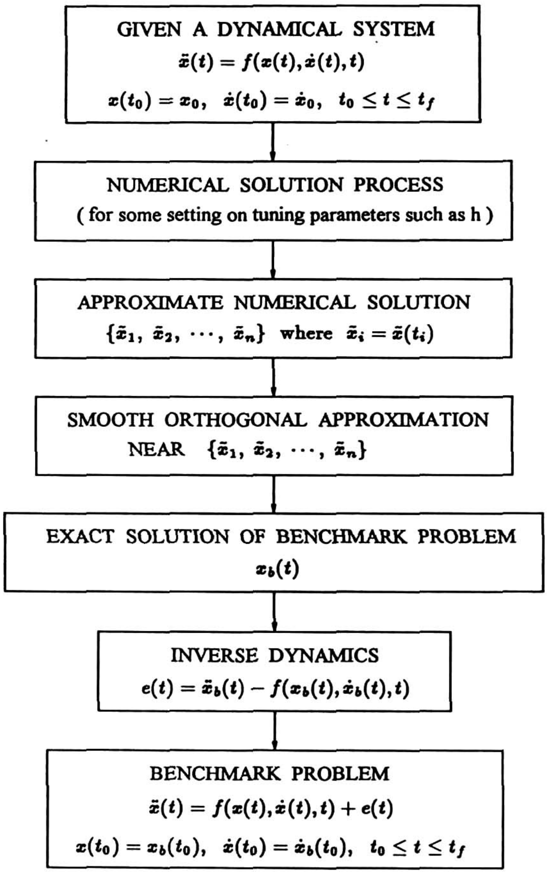

Fig. 1. Flow Chart for Construction of a Benchmark Problem

Now Eq.(4) is a benchmark problem neighboring Eq.(1) and we have arranged that $x(t)$, $\dot{x}(t)$ satisfy Eq.(4) exactly; and Eq.(5) becomes the benchmark problem neighboring Eq.(2) and we have arranged that $x(t)$, $\dot{x}(t)$, $\ddot{x}(t)$ satisfies Eq.(5) exactly. We obviously want the perturbation function $e(t)$ to be as small as possible, that is, the benchmark problem is not only a near neighbor of the original discrete solution, but it also very nearly satisfies the same differential equations. The previously discussed least square approximation method typically gives the poorest approximation near the ends of the interval. This may result in a relatively large $e(t)$ near the initial and final times. To avoid this problem we can integrate Eqs.(1) and (2) over the enlarged interval $t_{0^-} \le t \le t_{f^+}$ (where $t_{0^-} < t_0$, $t_{f^+} > t_f$) and use these numerical results as generators for analytical solutions over the original interval ($t_0 \le t \le t_f$). Experience indicates that a 20% "enlargement" $\{(t_{f^+} - t_{0^-}) \ge 1.2(t_f - t_0)\}$ is almost always sufficient to support good interpolation over the original interval ($t_0 \le t \le t_f$). If the measure of $e(t)$ is judged too large then we increase the number of Chebyshev polynomials m to reduce $e(t)$ over the whole interval, or "start over" by attempting to find a better approximate numerical solution to initiate the process. Figures 1 and 2 provide logical flow charts showing construction of a benchmark problem and an associated convergence study for second order systems.

Until now, we consider ODEs case. Analogous approach can be applied to PDEs which have two independent variables. Least square approximation associated with using the discrete version of the Chebyshev polynomials can be invoked to obtain the smooth $f(x, y)$ solution from the discrete solution.

Let's consider $n' \times m'$ discrete data points such as

$$z_{11} = f(x_1, y_1),\quad z_{12} = f(x_1, y_2),\ \cdots,\ z_{1m'} = f(x_1, y_{m'})$$

$$z_{21} = f(x_2, y_1),\quad z_{22} = f(x_2, y_2),\ \cdots,\ z_{2m'} = f(x_2, y_{m'})$$

$$\vdots$$

$$z_{n'1} = f(x_{n'}, y_1),\quad z_{n'} = f(x_{n'2}, y_2),\ \cdots,\ z_{n'm'} = f(x_{n'}, y_{m'})$$

where x_i, y_j are equally spaced independent variables.

How can we reliably compute a continuous, differentiable, analytical function f from the

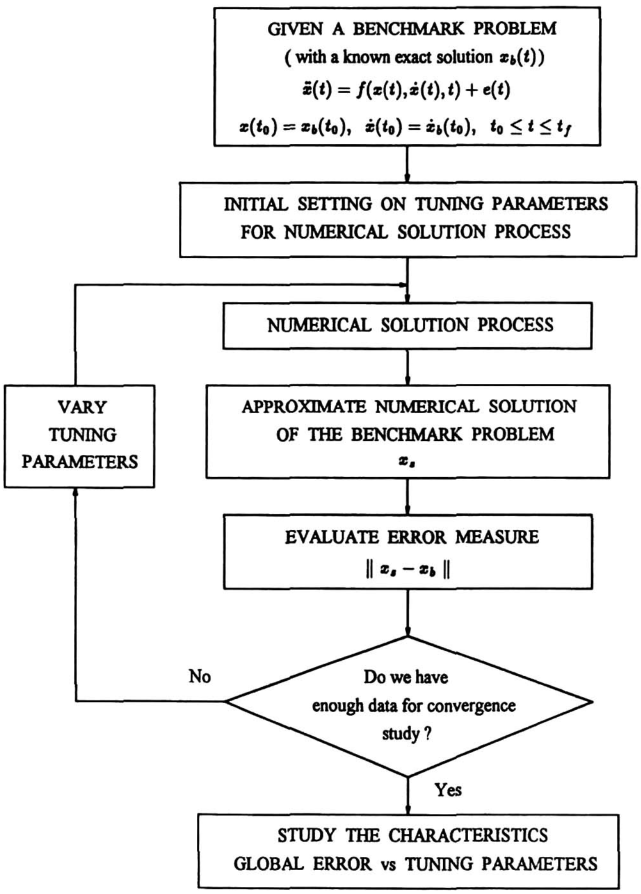

Fig. 2. Flow Chart for Convergence Study

data points in the least-squares sense? Analogous to the ODE case, we elect to make use of discrete orthogonality. We nondimensionalize (x, y) using

$$\bar{x}(x) = \frac{x - x_1}{h_x}, \qquad \bar{y}(y) = \frac{y - y_1}{h_y}$$

where h_x, h_y are the increments of x and y, respectively:

$$z = f(x, y) = F(\bar{x}, \bar{y})$$

From two-dimensional $n' \times m'$ data points, the function F can be approximated by $p \times q$ two-dimensional basis functions that come from the discrete version of the Chebyshev polynomials [weight function $w(x) = 1$] as follows:

$$F(\bar{x}, \bar{y}) \equiv \sum_{i=1}^{p} \sum_{j=1}^{q} b_{ij}\, T_i(\bar{x})\, T_j(\bar{y})$$

where $p \le n'$, $q \le m'$ and $T_*(*)$ is the univariate Chebyshev polynomial in the discrete range [12]. We use the conventional definitions of Chebyshev polynomials and the corresponding recurrence relations. Using discrete orthogonality properties of Chebyshev polynomials, the typical coefficient b_{rs} can be obtained as

$$b_{rs} = \frac{\sum_{i=1}^{n'} \sum_{j=1}^{m'} z_{ij}\, T_r(\bar{x}_i)\, T_s(\bar{y}_j)}{\sum_{i=1}^{n'} \sum_{j=1}^{m'} T_r(\bar{x}_i)\, T_s(\bar{y}_j)\, T_r(\bar{x}_i)\, T_s(\bar{y}_j)}$$

where $1 \le r \le p$, $1 \le s \le q$.

We can find $f(x, y)$ from $F(\bar{x}, \bar{y})$, since $f(x, y) = F(\bar{x}(x), \bar{y}(y))$. Using the above approximation method, we interpolate a smooth differentiable function as a two-variable orthogonal function expansion that passes near the $n' \times m'$ discrete data points.

Illustrative Examples

Now we demonstrate the previous ideas using four initial value problems for nonlinear differential equations. First we show the utility of the computer codes [14] for a simple nonstiff problems. Then, a celestial mechanics problem is introduced to illustrate the utility of this methodology when we use the IMSL subroutines DIVPRK and DIVPBS. And then we consider an idealized three-body distributed-parameter system which can be described by a hybrid ODE/PDE system. Finally, we consider a stiff problem in the fourth example.

Second Order Systems

We consider the following nonlinear, nonautonomous second order differential equation.

$$\ddot{x} = -x - 0.1(1+x^2)\,\dot{x} + 0.1x^3 + \sin 3t \tag{6}$$

where $x(0) = 1$ and $\dot{x}(0) = 0$, and we seek the solution over the interval $0 \leq t \leq 10$. We convert Eq.(6) to a first order system as follows:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - 0.1(1+x_1^{\,2})x_2 + 0.1\,x_1^{\,3} + \sin 3t \end{aligned} \tag{7}$$

where $x_1(0) = 1$ and $x_2(0) = 0$.

We solve Eq.(7) using the Runge-Kutta fourth order method to evaluate the candidate discrete approximate solution. Here we construct the interpolated solution using 121 data points over the 20% enlarged time interval $-1 \leq t \leq 11$. An analytical expression for $x_1(t)$ is obtained from the discrete approximate solution. In this problem, a degree 30 Chebyshev polynomial is established by the least square approximation. Substituting $x_1(t)$, $\dot{x}_1(t)$, $\ddot{x}_1(t)$ into Eq.(6) we calculate the function $e(t)$ that satisfies the following equation exactly.

$$\ddot{x} = -x - 0.1(1+x^2)\,\dot{x} + 0.1x^3 + \sin 3t + e \tag{8}$$

To use the Runge-Kutta method, Eq.(8) can be converted to a first order system as follows:

$$\begin{aligned} \dot{x}_1 &= x^2 \\ \dot{x}_2 &= -x_1 - 0.1(1+x_1^{\,2})x_2 + 0.1\,x_1^{\,3} + \sin 3t + e \end{aligned} \tag{9}$$

Now, Eq.(8) becomes a benchmark problem for Eq.(6), and $x(t)$ is an algebraic function that satisfies Eq.(8) exactly. When we use the pointwise error in the root mean square sense, Fig. 3 shows the relationship between global error and step size in log/log scale. The rate of convergence is 4 in this problem and this coincides with the fact that an rth order method should have a global error of $O(h^r)$ in the absence of arithmetic errors [5]. Figure 4 shows the perturbation term over the time interval. The critical value for step size is about 0.001. Now we consider the original problem. The relationship between step size and error at $t = 10$ is shown in Fig. 5 when we follow the common way assessing the true solution using the IMSL subroutines DIVPRK and DIVPBS. Comparing Figs. 3 and 5, we observe that the critical value h and the accuracy are almost the same.

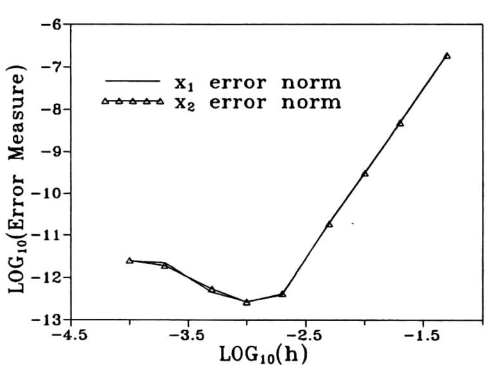

Fig. 3. Global Error vs. Step size for the Benchmark Problem

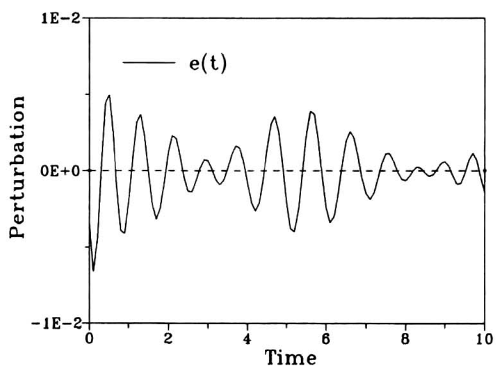

Fig. 4. Perturbation Term of the Second Order System

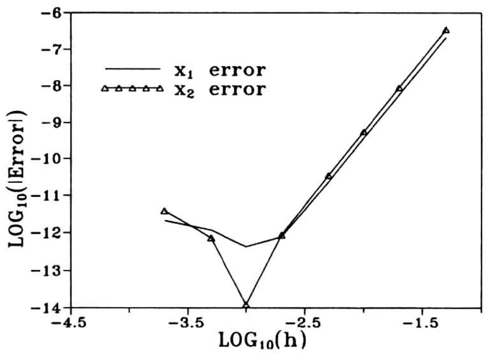

Fig. 5. Error (at t=10) vs. Step size for the Original Problem

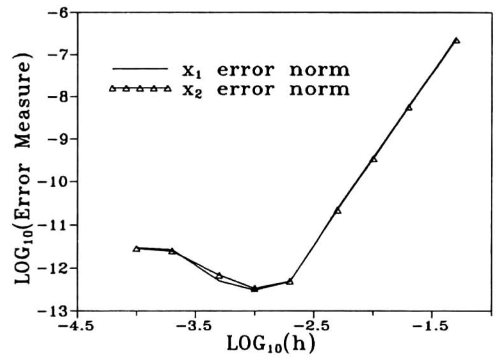

Fig. 6. Global Error vs. Step size for the Benchmark Problem of 20% perturbation

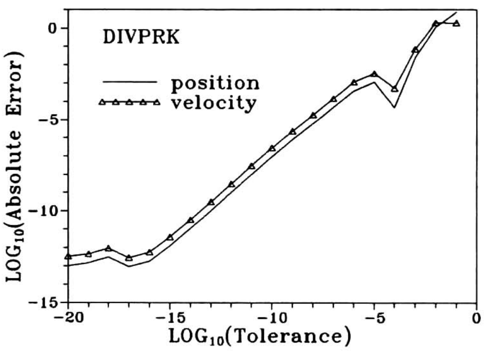

Fig. 7. Absolute Error vs. Tolerance for the Benchmark Problem(DIVPRK)

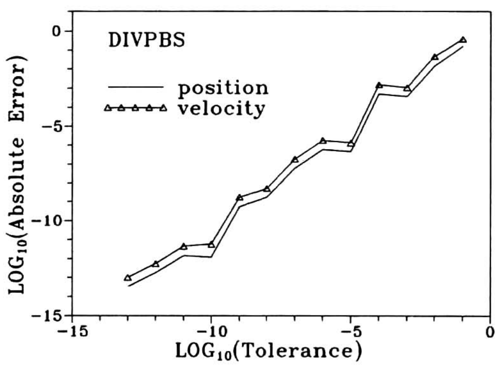

Fig. 8. Absolute Error vs. Tolerance for the Benchmark Problem(DIVPBS)

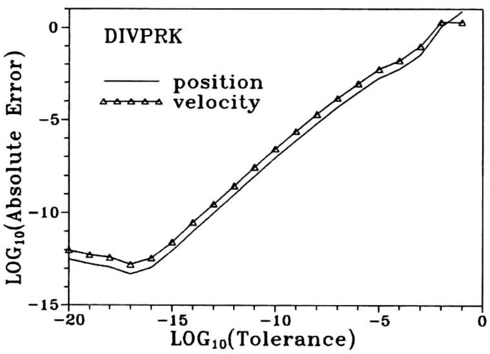

Fig. 9. Absolute Error vs. Tolerance for the Two Body Problem(DIVPRK)

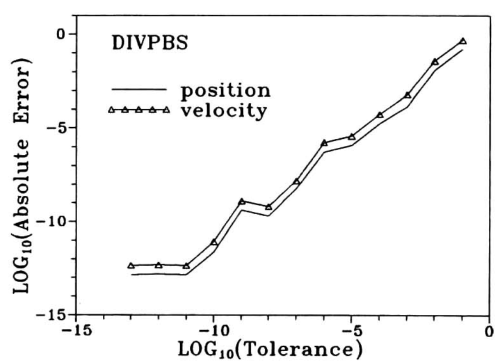

Fig. 10. Absolute Error vs. Tolerance for the Two Body Problem(DIVPBS)

We change the initial conditions slightly and the nonautonomous term in the differential equation as follows:

$$\ddot{x} = -x - 0.1(1 + x^2)\dot{x} + 0.1x^3 + 1.2\sin 3t \tag{10}$$

where $x(0) = 1.2$ and $\dot{x}(0) = 0.2$ over the interval $0 \le t \le 10$.

After using the same procedure, we obtain the global error/step size relationship shown in Fig. 6. We notice that Figs. 3 and 6 are almost the same. In other words, the critical value for h and the accuracy are almost identical even though there are 20% perturbations in the differential equation, in this case.

Two Body Problem

We consider the simple two body problem. The exact solution is periodic with period 2π and the solution traces out an ellipse with eccentricity 0.6.

$$\begin{aligned} \ddot{x} &= -x/r^3, \quad x(0) = 0.4, \quad \dot{x}(0) = 0 \\ \ddot{y} &= -y/r^3, \quad y(0) = 0, \quad \dot{y}(0) = 2 \end{aligned} \tag{11}$$

where $r = (x^2 + y^2)^{1/2}$.

These equations can be solved exactly and we can find the analytical solution in Ref. [15]. We reformulate Eq.(11) as a first order system as follows:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1/\,(x_1^2+x_3^2)^{3/2} \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -x_3/\,(x_1^2+x_3^2)^{3/2} \end{aligned} \tag{12}$$

where

$$x_1(0)=0.4, \quad x_2(0)=0, \quad x_3(0)=0, \quad x_4(0)=2\,.$$

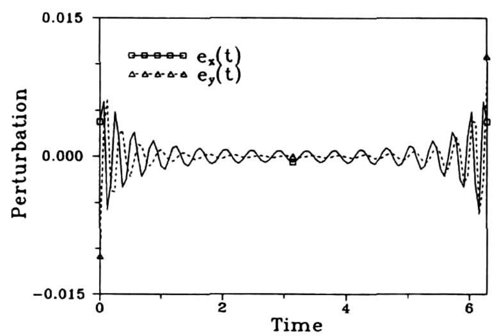

Fig. 11. Perturbation Terms of the Two Body Problem

We solve Eq.(12) using DIVPRK to evaluate the candidate discrete approximate solution. Here we use 121 data points over the 20% enlarged time interval and a degree 50 Chebyshev polynomial approximation is used for the least square fitting of $x_1(t)$ and $x_3(t)$. After constructing the benchmark problem, we do an absolute error test on $(0, 2\pi)$. Figures 7 and 8 show the relationship between absolute error and tolerance in log/log scale when we use DIVPRK and DIVPBS for the benchmark problem. Figures 9 and 10 show the relationship between absolute error and tolerance in log/log scale when we use DIVPRK and DIVPBS for the original two body problem. We notice that Figs. 7 and 8 are almost identical to Figs. 9 and 10, respectively. The perturbation terms are shown in Fig. 11. Thus the benchmark problem (constructed by the method of this study) essentially gives results that are identical to those obtained by using the exact solution of the original problem.

Three-Body Distributed-Parameter System

With reference to Fig. 12, we consider a rigid hub with a cantilevered flexible appendage that has a finite tip mass. Table 1 summarizes the configuration parameters of this flexible structure. The appendage is considered to be a uniform flexible beam, and we make the Euler-Bernoulli assumptions of negligible shear deformation and negligible distributed rotatory

Table 1. Configuration Parameters of Three-Body Problem

PARAMETER	SYMBOL	VALUE
Hub radius	r	1 ft
Rotary inertia of hub	J_h	8 $slug \cdot ft^2$
Mass density of beam	ρ	0.0271875 $slug/ft$
Elastic modulus of beam	E	0.1584×10^{10} lb/ft^2
Beam length	L	4 ft
Moment of inertia of beam	I	$0.47095028 \times 10^{-7}$ ft^4
Tip mass	m_t	0.256941 $slug$
Rotary inertia of tip mass	J_t	0.0028 $slug \cdot ft^2$

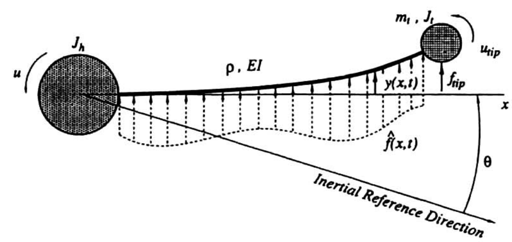

Fig. 12. Three-Body Distributed-Parameter System

inertia. The beam is cantilevered rigidly to the hub. Motion is restricted to the horizontal plane, and we neglect the velocity component $-y\dot{\theta}$ that is perpendicular to the y direction. The control system is assumed to generate a torque u acting upon the hub, a torque u_{tip} and a force f_{tip} acting upon the tip mass, and a distributed force density $\hat{f}$ acting upon the appendage. We assume small elastic motions viewed from the hub-fixed rotating reference frame. Overdots denote derivatives with respect to time and primes denote derivatives with respect to the spatial position.

Using an explicit version of the classical Lagrange equation for hybrid coordinate distributed-parameter systems [16], the governing differential equations and the boundary conditions are obtained efficiently:

$$J_h \ddot{\theta} + \int_0^L \rho(x+r)[\ddot{y}+(x+r)\ddot{\theta}]\,dx + m_t(L+r)[(L+r)\ddot{\theta}+\ddot{y}(L)] + J_t[\ddot{\theta}+\ddot{y}'(L)]$$
$$= u + \int_0^L \hat{f}(x)(x+r)dx + (L+r)f_{tip} + u_{tip} \tag{13}$$

$$\rho[\ddot{y}+(x+r)\ddot{\theta}] + EIy'''' = \hat{f} \tag{14}$$

$$EI\left.\frac{\partial^3 y}{\partial x^3}\right|_L - m_t[(L+r)\ddot{\theta}+\ddot{y}(L)] + f_{tip} = 0 \tag{15}$$

$$EI\left.\frac{\partial^2 y}{\partial x^2}\right|_L + J_t[\ddot{\theta}+\ddot{y}'(L)] - u_{tip} = 0 \tag{16}$$

Notice that if we knew an explicit, differentiable solution for the motion variables $\{y(x,t), \theta(t)\}$, then Eqs.(13-16) can be solved directly and exactly for the four corresponding time and space varying forces and moments $\{u(t), \hat{f}(x,t), u_{tip}(t), f_{tip}(t)\}$, thus yielding the desired inverse solution. We construct an exact solution that is near neighbor of a given approximate solution. We are interested in physically meaningful problem. Therefore we use finite element method to obtain linear finite dimensional equations of motion for the model. And we design a typical control law using the LQR, and modal coordinates are used to design controller. The optimal feedback control is obtained by solving the Riccati equation [17].

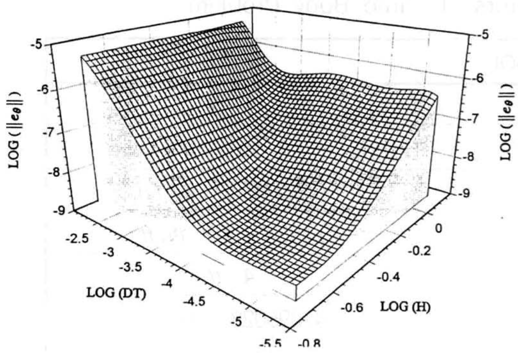

Fig. 13. Error Norm Distribution of θ for the Example 3

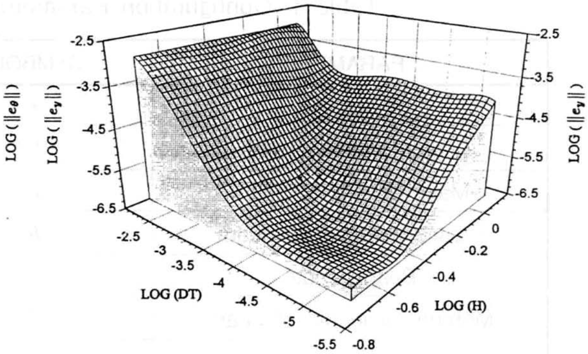

Fig. 14. Error Norm Distribution of y for the Example 3

We can solve the initial value problem using a time discretization process and then we obtain $\tilde{y}(x_i, t_i), \tilde{\theta}(t_i)$, and $\tilde{u}(t_i)$ at discrete points in space and time for the enlarged time interval $(0 \le t \le 0.1)$. The initial condition for θ is 0.1 rad, and the third natural mode of this flexible structure is used for the initial deformation $y(x,0)$. Here we construct a benchmark problem from the previous candidate approximate solutions for time interval $(0 \le t \le 0.08)$. We have an analytical set $y_b(x,t), \theta_b(t)$, and $\{u(t), \mathcal{F}(x,t), u_{tip}(t), f_{tip}(t)\}$ that satisfy Eqs.(13-16) exactly. Given initial conditions $\{y(x,0) = y_b(x,0),\ \theta(0) = \theta_b(0)\}$ and force functions $\{u(t), \mathcal{F}(x,t), u_{tip}(t), f_{tip}(t)\}$, the approximate simulation of this structure's dynamics $\{y_s(x,t), \theta_s(t)\}$ can proceed. When we use the Newmark integration method with finite element modeling, the convergence and accuracy behavior is studied as a function of the number of finite elements and the integration step size. The error norm distribution of θ and y is shown in Figs. 13 and 14, respectively, as a function of time step size(DT) and mesh size(H).

Here we introduce the following definitions for the supmetric error:

$$\|e_\theta(t)\|_{L^2(0,T)} \equiv \left[\int_0^T e_\theta(t)^2\, dt\right]^{1/2}$$

$$\|e_y(x,t)\|_{L^2(0,T:L^2)} \equiv \left[\int_0^T \int_0^L e_y(x,t)^2\, dx\, dt\right]^{1/2}$$

where $e_\theta(t) = \theta_s(t) - \theta_b(t)$, $e_y(x,t) = y_s(x,t) - y_b(x,t)$.

A Stiff Problem

We consider the following problem [18] that represents a typical stiff problem. Although this problem is expressed by linear ODE instead of nonlinear, it shows the stiff behavior very well.

$$\begin{aligned} \dot{x}_1 &= -29998\,x_1 - 39996\,x_2 \\ \dot{x}_2 &= 14998.5\,x_1 + 19997\,x_2 \end{aligned} \tag{17}$$

where $x_1(0) = 1,\ x_2(0) = 1$.

The exact solutions of Eq.(17) are as follows:

$$\begin{aligned} x_1(t) &= 7\exp(-10^4 t) - 6\exp(-t) \\ x_2(t) &= -3.5\exp(-10^4 t) + 4.5\exp(-t) \end{aligned} \tag{18}$$

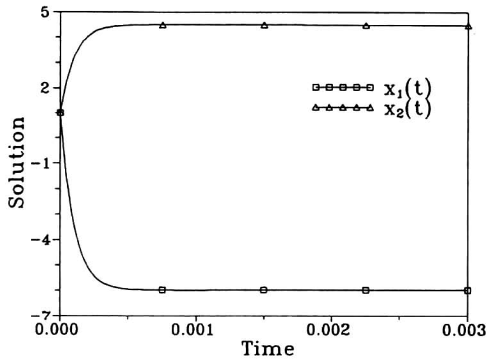

Fig. 15. Solution of the Example 4 for the Rapid Change Region

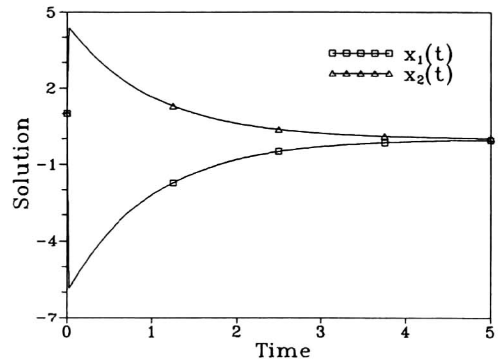

Fig. 16. Solution of the Example 4 for the Gradual Change Region

The eigenvalues of the coefficient matrix are -1 and -10^4. Figures 15 and 16 show the solutions over two different intervals, a region of very rapid change followed by gradual asymptotic behavior. It is almost impossible to obtain a satisfactory orthogonal function benchmark problem that covers both regions with a reasonable number of terms. We conclude that the proposed methodology is not adequate for such stiff problems unless piecewise approximation methods, for example, the type introduced by Junkins et al. [19] are used. Stiff problems are relatively expensive to solve and the expense depends strongly on the tolerance [5,18]. Enright et al. [20] provide a good collection of stiff test problems.

Summary and Conclusion

An inverse method is introduced by first considering the initial value problem for nonlinear ordinary differential equations. Used as an exact benchmark solution for a numerical convergence study, this methodology gives valuable information about the optimal tuning parameters, and the accuracy of the numerical solution process for a class of ODE problems and for a given solution code. The inverse idea introduced for ODE systems is generalized to accomodate construction of benchmark problems for hybrid ODE/PDE systems. A multi-variable orthogonal function expansion method and computer symbol manipulation are successfully used for this process. This methodology makes it possible for one to rigorously determine exact solution errors and to study the convergence and accuracy behavior as a function of tuning parameters for a class of ODE/PDE systems for which the initial value problem is not exactly solvable. We present numerical examples to explore the practical utility of this approach for both ODEs and hybrid ODE/PDE systems. In the second example, we show the utility of this methodology using the IMSL subroutines DIVPRK and DIVPBS as solvers. We investigate the absolute error/tolerance relationship and compare DIVPRK and DIVPBS. And in the third example, we demonstrate an inverse dynamics method for constructing exact special case solutions for hybrid ODE/PDE systems. Numerical examples indicate that a rigorous error analysis is obtained not merely for one nominal solution, but for a substantial neighborhood of the nominal solution. By constructing a family of neighboring benchmark problems, one can obtain valuable information about the convergence and accuracy properties that are relatively invariant with respect to perturbations within a known bound.

References

1. Suk, J., and Kim, Y., "On the Modeling of Dynamic Systems," *KSAS International*

Journal, Vol. 2, No. 1, 2001, pp. 78-92.

2. Miele, A., and Iyer, R. R., "General Technique for Solving Nonlinear, Two-Point Boundary-Value Problems via the Method of Particular Solutions," *Journal of Optimization Theory and Applications*, Vol. 5, No. 5, 1970, pp. 382-403.

3. Silver, W. M., "On the Equivalence of Lagrangian and Newton-Euler Dynamics for Manipulators," *The International Journal of Robotics Research*, Vol. 1, No. 2, 1982, pp. 60-70.

4. Luh, J. Y. S., Walker, M. W., and Paul, R. P. C., "On-Line Computational Scheme for Mechanical Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 102, 1980, pp. 69-76.

5. Gear, C. W., *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1971.

6. Shampine, L. F., "Limiting Precision in Differential Equation Solvers, II: Sources of Trouble and Starting a Code," *Math. Comp.*, Vol. 32, No. 144, 1978, pp. 1115-1122.

7. Enright, W. H., "Analysis of Error Control Strategies for Continuous Runge-Kutta Methods," *SIAM J. Numer. Anal.*, Vol. 26, No. 3, 1989, pp. 588-599.

8. Hull, T. E., Enright, W. H., Fellen, B. M., and Sedgwick, A. E., "Comparing Numerical Methods for Ordinary Differential Equations," *SIAM J. Numer. Anal.*, Vol. 9, No. 4, 1972, pp. 603-637.

9. Krogh, F. T., "On Testing a Subroutine for the Numerical Integration of Ordinary Differential Equations," *Journal of the Association for Computing Machinery*, Vol. 20, No. 4, 1973, pp. 545-562.

10. *IMSL MATH/LIBRARY User's' Manual Version 1.1*, IMSL Inc., 1989.

11. Junkins, J. L., *An Introduction to Optimal Estimation of Dynamical Systems*, Sijhoff & Noordhoff, Alphen aan den Rijn, The Netherlands, 1978.

12. Abramowitz, M., and Stegun, I. A., *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, National Bureau of Standards, Applied Mathematics Series 55, U.S. Department of Commerce, 1972.

13. *MACSYMA Reference Manual Version 13*, Symbolics Inc., 1988.

14. Junkins, J. L., and Lee, S., "Benchmark Problems for the Solution of Ordinary Differential Equations," *Shock and Vibration Computer Programs*, Vol SVM-13, 1995, pp. 497-523.

15. Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics*, AIAA Education Series, New York, New York, 1987.

16. Lee, S., and Junkins, J. L., "Explicit Generalization of Lagrange's Equations for Hybrid Coordinate Dynamical Systems," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 6, 1992, pp. 1443-1452.

17. Junkins, J. L., and Kim, Y., *An Introduction to Dynamics and Control of Flexible Structures*, AIAA, Washington, DC, 1993.

18. Shampine, L. F., and Gordon, M. K., *Computer solution of Ordinary Differential Equations*, W. H. Freeman, San Francisco, 1975.

19. Junkins, J. L., Miller, G. W., and Jancaitis, J. R., "A Weighting Function Approach to Modeling of Irregular Surfaces," *Journal of Geophysical Research*, Vol. 78, No. 11, 1973, pp. 1794-1803.

20. Enright, W. H., Hull, T. E., and Lindberg, B., "Comparing Numerical Methods for Stiff Systems of ODEs," *BIT*, Vol. 15, 1975, pp. 10-48.