Journal of the Nigerian Association of Mathematical Physics
Volume **25 (November, 2013), pp 51 – 54**

Error Control of Initial Value Problems for Stiff Ordinary Differential Equations Using Neumaier's Method

George N. Emenogu

Department of Mathematics
Michael Okpara University of Agriculture,
Umudike, Abia State, Nigeria

Abstract

Validated methods when compared with standard numerical methods for initial value problems (IVPs) for ordinary differential equations (ODES) not only compute a numerical solution to a problem, but also generate guaranteed global error bounds associated with the numerical solution. There have been significant developments in the field of validated numerical methods for IVPs over the past few decades. However, none of the validated methods develop to date are suitable for stiff problems. This paper investigated the potential of Neumaier's validated methods for solving stiff IVPs for ODES. Neumaier's result which is a special case of Dahlquist's [1] result shows from our findings an effective validated method for stiff IVPs for ODES, for problems where there is no wrapping effect.

1.0 Introduction

Obviously, a good number of differential equations does not have an analytical solution, but with the advent of digital computers, complex equations or systems of equations could be solved with various numerical methods. Probably, the most serious drawback of numerical methods is that they can only approximate the continuous solution with a series of discrete points. A large number of formations were developed to solve these kinds of equations. Implicit Adam-Bashforth and explicit Runge-Kutta methods are used fairly extensively nowadays.

Now consider the initial value problem (IVP) for ordinary differential equation (ODE)

$$y'(t) = f(t, y(t)), y(t_0) = y_0, \qquad t\epsilon\,(t_0, T) \tag{1}$$

Where $y\ \epsilon R^n\ and\ f$: $\quad R \times R^n \rightarrow R^n$

We denote the set of n-dimensional real interval vectors by R^nassume that f is smooth, $(i.e.f\ is\ differentiable)$ at all points of the curve, and there exists a unique solution to equation (1.1) on (t_0, T). The purpose of this paper is to investigate the potential of Neumaier's [2] enclosure method for the solution of the IVP (1.1) and to provide an insight into how this method behaves in practices.

2.0 Test Problems and Numerical Results

In this section, four 2-Dimensional initial value problems (IVP) are presented. The numerical results obtained for these test problems and the shortcomings that they reveal in our implementation of Neumaier's enclosure method are discussed.

2.1 Test Problems

Listed below are the four 2-dimensional test problems, each of the form

$$y'(t) = A(t) + b(t), \qquad where\ t\epsilon\,[0, T].$$

The global error tolerance $\tau\ is\ set\ to\ 10^{-6}$, a representative value for this parameter

The initial condition for each test problem is

$$\begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Problem 1

Linear ODE with constant coefficients and zero

$$b(t), t\epsilon\,[0, 100], \mu_k = U_k^n = -1\ for\ all\ k$$

Corresponding author: ***George N. Emenogu,*** E-mail: gnemenogu@hotmail.com, Tel.: +234 7069124533

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1000 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (2.1)$$

The true solution is $y_1(t) = e^{-t}$ *and* $y_2(t) = e^{-103t}$

Problem 2

Linear ODE with constant coefficients and constant non-zero $b(t), t\epsilon\,[0,100]$, $\mu_k = U_k^n = -1\ for\ all\ k$:

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1000 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (2.2)$$

The true solution is $y_1(t) = 1$ *and* $y_2(t) = \frac{1}{10^3} + \frac{999}{10^3} e^{-103t}$

Problem 3

Linear ODE with time dependent coefficients and $b(t)\ \ t\epsilon\,[0, 10\pi], \mu_k$ *and*

$$\mu_k\ \epsilon\left(-10, -10 + \frac{14}{\sqrt{2}}\right) \approx [-10, 0.1]\ for\ all\ k:$$

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{pmatrix} -10 & 14cos(t) \\ 14sin(t) & -10 \end{pmatrix} \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} + \begin{pmatrix} e^{-3\sin(t)} \\ t \end{pmatrix} \quad (2.3)$$

Problem 4

Linear ODE with constant $A(t)$ and time dependent $b(t)$ having a spike at $t = 1, t\epsilon\,[0,2], \mu_k = \widehat{U}_k = -1\ for\ all\ k$:

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} + \begin{bmatrix} \dfrac{(t-2)^2 + 0.000001^2 - 1}{((t-1)^2 + 0.00001^2)} \\ 0 \end{bmatrix} \quad (2.4)$$

The true solution is

$$y_1(t) = \frac{10^{10}}{10^2 t^2 - 2 \times 10^{10} t + 10000000001} + \frac{1}{10000000001} e^{-t}$$

And $y_2(t) = e^{-t}$

2.2 numerical results and discussion

The number of significant digits of precision that MAPLE uses can be set using the command digits. In our implementation, we set digits to 15. Given a user-specified global error bound τ. we chose the largest step size h_k at each time-step, h_k such hat

$$\emptyset_k e^{\mu_k h_k} + \frac{\epsilon_k^n}{\mu_k^n}(e^{\mu_k h_k} - 1) \le \tau \quad (2.5)$$

And

$$\frac{h_{k-1}}{20^l} \le 10\ h_{k-1}, for\ l = 0, \ldots, 10, \qquad are\ satisfied$$

The following are the graphs obtained for problems 1, 2, 3, and 4 using the above step size control strategy. The step sizes taken for the problems are presented in tables 2.1, 2.2 and 2.3 respectively.

Note that at the last integration step, the step size chosen can be smaller than if the function is to be evaluated over the time interval $t\,\epsilon\,[0,T]\ where\ \tau > T$. thus by taking a smaller step size at the last integration step results in a smaller error bound at the last integration step.

We estimate $\propto_k = \|S_k^{-1} S_{k-1}\| \emptyset_{k-1}$

Where S_k *is the eigenvector matrix associated with* A_k.

$$\text{let} C = \|S_k^{-1} S_{k-1}\| \quad (2.6)$$

For problems 1, 2 and 4, the matrix A is fixed with $\mu_k = -1\ for\ all\ k$ this implies that $C = 1, since\ S_k = s\ for\ all\ k\ and\ so\ S_k^{-1} S_{k-1}$ is the identity matrix. Hence the value of $\propto_k$ for integration time-step $(k > 1)$. This is true for any fixed matrix A, since C is always equal to 1. Thus for problem with $\mu_k < 0$ and with fixed eigenvectors, the integration can always be continued while keeping the global error $\le \tau$.

For problem 4, the matrix A_k is time dependent and this implies that the matrix S_k may change with k. It was noted that although the value of $\emptyset_{k-1}$ at the end point integration time-step $(k-1)$ may be below the prescribed error bound, the initial value of $\emptyset_k$ at integration time-step k may exceed the prescribed error bound. This occurs because the value of C may exceed 1 on some steps, due to S_k changing from step to step. As a result, the value of $\propto_k$ at integration step k may exceed the final value of $\emptyset_{k-1}$ at integration step $(k-1)$. Thus, if the eigenvectors are changing from step to step, there is possibility that the integration error will exceed the maximum tolerated error τ. In other words, depending on the value of C, the initial value of $\emptyset_k$ could exceed τ, even if the final value of $\emptyset_{k-1}$ in the previous integration time-step is below τ. This occurred in problem 4.

For problem 4, the following output was observed:

- At integration time-step $k = 1$

$$\emptyset_{final} = 0.914080293343665 \times 10^{-6}$$

- At integration time-step $k = 2$

$$C = 1.46096064526068.$$

$$\emptyset_{Initial} = 1.28496606987386 \times 10^{-6},$$

Which is greater than the prescribed error bound

3.0 Conclusion

When the integration fails to continue using the step-size control strategy described previously, Neher [3,4] proposed an alternate condition for the step size control in order for the integration to continue. This alternate condition allows the global error to exceed the prescribe tolerance τ, but by an amount such that the global error would not grow too fast. More precisely, it allows the global error bound at the end of the integration step k to grow by at most x times the damped initial error at the end of integration step k, where $1 \leq x \leq 1\frac{1}{3}$. that is, the integration can continue provided that

$$\emptyset_k(h_k) \leq \tau) \; or \; \left|\frac{\hat{\xi}_k}{\hat{U}_k}\left(e^{\hat{U}_k h_k} - 1\right)\right| < \left|\frac{\propto_k e^{\mu_k^n h_k}}{C}\right| \qquad (2.7)$$

Where C is a positive integer. If $C = 3$, then the global error bound at the end of integration step k is at most $1\frac{1}{3}$ times the damped initial error bound at integration step k.

Table 1: Stepsize for each integration step for problem 1,

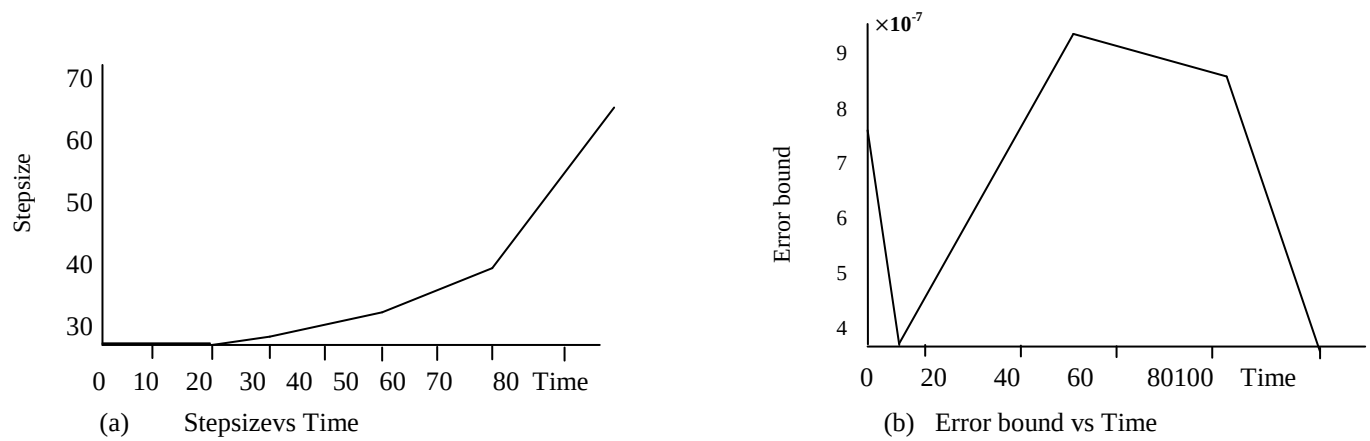
Integration step k	**Initial timestept_{k-1}**	**Final timestept_k**	**Stepsize taken at $h_k = t_{k-1}$**	**Error $e(t) = p(t) - y(t)$**
1	0	0.0125	0.0125	0.0000006
2	0.0125	0.06625	0.05375	0.0000005
3	0.06625	0.60375	0.5375	0.0000004
4	0.60375	5.00912	4.40537	0.0000001
5	5.00912	5.97875	0.96963	0.0000009
6	5.97875	20.66	14.68125	0.0000008
7	20.66	33.66	13.0000	0.0000002
8	33.66	100	66.3400	0.0000001

Table 2: Stepsize for each integration step for problem 2

Integration step k	**Initial timestept_{k-1}**	**Final timestept_k**	**Stepsize taken at $h_k = t_{k-1}$**	**Error $e(t) = p(t) - y(t)$**
1	0	0.0125	0.0125	0.00000065
2	0.0125	0.004375	0.03125	0.00000012
3	0.004375	0.33212	0.28837	0.0000002
4	0.33212	0.35625	0.02413	0.0000001
5	0.35625	3.48125	3.125	0.0000001
6	3.48125	21.73125	18.2500	0.0000001
7	21.73125	34.73125	13.00125	0.0000001
8	34.73125	100	65.26875	0.0000001

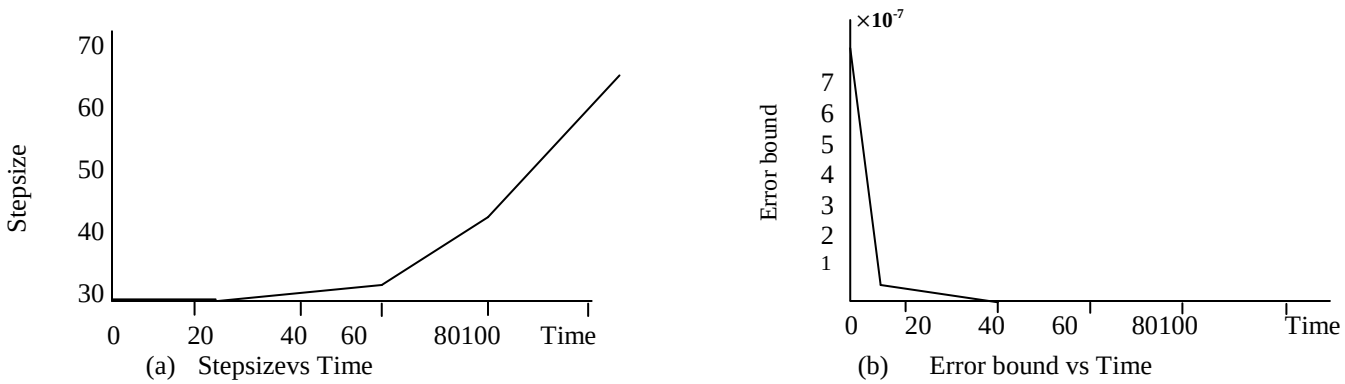
Table 3:Stepsize for each integration step for problem 3

Integration step k	**Initial timestept_{k-1}**	**Final timestept_k**	**Stepsize taken at $h_k = t_{k-1}$**	**Error $e(t) = p(t) - y(t)$**
1	0	0.0125	0.0125	0.0000007
2	0.0125	0.040625	0.028125	0.00000095
3	0.040625	0.321875	0.28125	0.0000008
4	0.321875	2.3046875	1.9828125	0.00000087
5	2.3046875	3.8909375	15.86245	0.0000008
6	3.8909375	7.9410575	4.05012	0.0000003
7	7.9410575	16.818875	8.8778175	0.0000002
8	16.818875	65.827838	49.008963	0.0000001
9	65.827838	76.2873875	10.459459	0.0000001
10	76.2873875	100	23.7126125	0.0000001



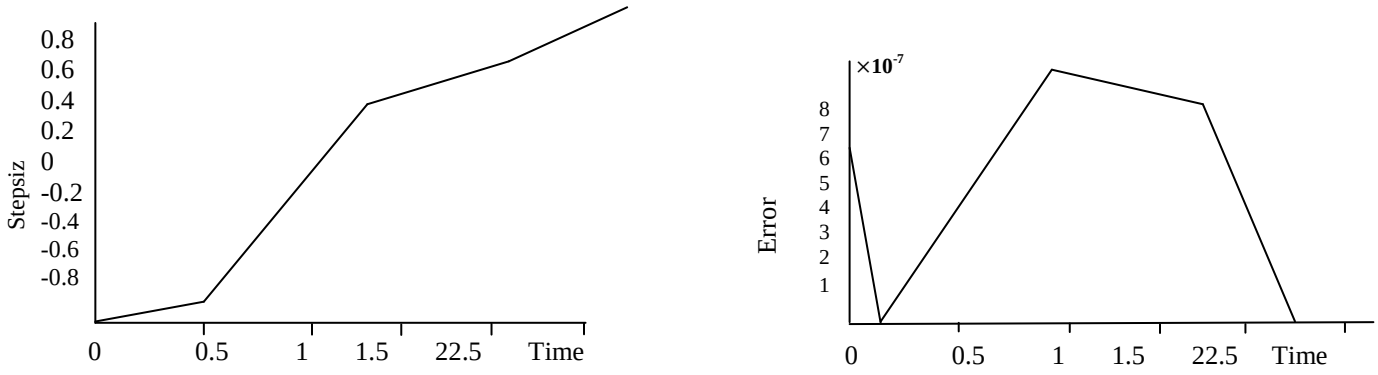
(a) Stepsizevs Time (b) Error bound vs Time

Figure 1: Problem 1: $\mu_k = \hat{\mu}_k = -1, digits = 15, integration\ steps\ taken = 8$



(a) Stepsizevs Time (b) Error bound vs Time

Fig. 2: Problem 6: $\mu_k = \hat{\mu}_k = -1, digits = 15, integration\ steps\ taken = 6$



(a) Stepsizevs Time (b) Error bound vs time

Figure 3: Problem 6: $\mu_k = \hat{\mu}_k = -1, digits = 30, integration\ steps\ taken = 3$

References

[1] G. Dahlquist, On stability and Error analysis for stiff non-linear problems, part 1, dept of comp. sci.Roy.Inst. of technology Report TRITA-NA-7508,(1975)

[2] A. Neumaier, Global, Rigorous and Realistic Bound for the solution of Dissipative Differential Equations. Part 1: Theory, computing Volume 52, No. 4: Page 315-336, 1994.

[3] M. Neher. On Neumaier's En closure method for the solution of Dissipative ODEs. Third int. Workshop on Taylor methods, Miami Beach,page 88-100, 2004.

[4] N. S. Nedialkov. Computing Rigorous Bound in the solution of an initial value problems for an ordinary Differential Equation. Ph.D Thesis. Department of Computer Science, University of Toronto, 1999.