



Applied Mathematics and Nonlinear Sciences

<https://www.sciendo.com>

A new computational algorithm for the solution of second order initial value problems in ordinary differential equations

P.K. Pandey[†]

Department of Mathematics, Dyal Singh College (Univ. of Delhi) Lodhi Road, New Delhi-110003, India

Submission Info

Communicated by Juan L.G. Guirao

Received 9th March 2018

Accepted 15th May 2018

Available online 15th May 2018

Abstract

In this article, we propose a new computational method for second order initial value problems in ordinary differential equations. The algorithm developed is based on a local representation of theoretical solution of the second order initial value problem by a non-linear interpolating function. Numerical examples are solved to ensure the computational performance of the algorithm for both linear and non-linear initial value problems. From the results we obtained, the algorithm can be said computationally efficient and effective.

Keywords: Absolute Stability, Finite difference approximations, Initial Value Problems, Maximum Absolute Error, Nonlinear IVP
AMS 2010 codes: 65L10, 65L12.

1 Introduction

Many phenomena that occur in chemical, biological, engineering, physical and social sciences can be modelled mathematically in the form of either ordinary or partial differential equations. However it is difficult to obtain exact solution for these differential equations especially if it is nonlinear, by analytical means. So we consider an approximate solution to these problems. There are numerous ways by which an approximate solution can be constructed. In numerical analysis a concept of approximation play very important role. Thus solving these practical problems which modelled as differential equation approximately, is one of the main preoccupations in numerical analysis.

Consider second order initial value problems in ordinary differential equations of the form

$$y^{(2)}(x) = f(x, y, y'), \quad x \in [a, b] \subset \mathbb{R} \quad \text{and} \quad y(x), y'(x), f(x, y, y') \in \mathbb{R}$$

(1.1)

[†]Corresponding author.

Email address: pramod_10p@hotmail.com

subject to initial conditions

$$y(a) = \alpha \quad \text{and} \quad y'(a) = \beta,$$

In the literature, problems of the form (1.1) are conventionally solved by reducing the differential system to first order equations. Some eminent authors have contributed in this specific area of research [1,2,3,4,11]. Another approach to investigate the solution of such problems were and referred to as shooting method either simple or multiple [8]. In recent years researchers [5,6] applied a nonstandard method and obtained competitive results to those obtained with other method. So, much research have reported on the numerical integration of initial value problems in literature, many of them are excellent work. But a concept to develop a new algorithm to solve equation (1.1) can not be over emphasized.

In this article, we develop a new single step algorithm capable of solving equations of the form (1.1). The similar algorithm was first reported [7] in study of first order initial value problems. Having seen the performance of the algorithm for solution of first order initial value problems, we are motivated and challenged to investigate what will happen if a similar idea is used to derive an algorithm for solution of second order initial value problems.

The existence and uniqueness of the solution to initial value problem (1.1) is assumed. Further we assume that problems (1.1) is well posed with continuous derivatives and that the solution depends differentially on the initial conditions. The specific assumption on $f(x, y, y')$ to ensure existence and uniqueness will not be considered [8,9,10].

This paper is divided into five sections. Section 2 deals with the derivation and development of the algorithm while truncation error and convergence of the algorithm are developed in Section 3. The stability of the algorithm is discussed in section 4 while numerical experiments on four model problems are presented in section 5.

2 Development of Algorithm

We define N , the finite number of the nodal points of the interval $[a, b]$, in which the solution of the problem (1.1) is desired as

$$x_j = a + jh, \quad j = 0, 1, 2, \dots, N \quad (2.2)$$

where the terms in right side of expression (2.2) are defined as,

the step length $h = \frac{(b-a)}{N}$, and $x_N = b$

Suppose we have to determine a number y_j , which is numerical approximation to the value of the theoretical solution $y(x)$ of problem (1.1) at the nodal point x_j , $j = 1, 2, \dots, N$ and other similarly notations like f_j defined as $f(x_j, y_j, y'_j)$. Following the ideas in [1,6], assuming the local assumption that the theoretical solution $y(x)$ to the initial value problem (1.1) can be locally represented in the interval $[x_j, x_{j+1}]$ by the interpolating function

$$F(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot e^{x^3} \quad (2.3)$$

where $a_0, a_1, a_2,$ and a_3 are undetermined coefficients.

To determine these undetermined coefficients, let impose these following conditions.

1. The interpolating function and its first derivative w.r.t. x must coincide with $y(x)$ and $y'(x)$ the theoretical solution and derivative of solution w.r.t. x of the problem (1.1) at $x = x_j$ and $x = x_{j+1}$ i.e.

$$\begin{aligned} F(x_j) &= y(x_j) \quad \text{and} \quad F(x_{j+1}) = y(x_{j+1}) \\ F'(x_j) &= y'(x_j) \quad \text{and} \quad F'(x_{j+1}) = y'(x_{j+1}) \end{aligned} \quad (2.4)$$

2.The second and third derivatives w.r.t x ,of the interpolating function respectively coincide with $f(x,y,y')$ and derivative of $f(x,y,y')$ w.r.t. x at $x = x_j$ i.e.

$$F^{(2)}(x_j) = f_j \quad \text{and} \quad F^{(3)}(x_j) = f'_j \tag{2.5}$$

Thus ,from assumptions (2.4,2.5) ,we will get

$$\begin{aligned} a_0 + a_1.x_j + a_2.x_j^2 + a_3.e^{x_j^3} &= \alpha \\ a_1 + 2.a_2.x_j + 3.a_3.x_j^2.e^{x_j^3} &= \beta \\ 2.a_2 + 3.a_3.x_j.(2 + 3.x_j^3).e^{x_j^3} &= f_j \\ 3.a_3.(2 + 18.x_j^3 + 9.x_j^6).e^{x_j^3} &= f'_j \end{aligned} \tag{2.6}$$

Solving the system of equation (2.6) for a_0, a_1, \dots ,we will obtain

$$\begin{aligned} a_1 &= \beta - x_j.f_j + \frac{(x_j^2 + 3x_j^5)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \\ a_2 &= \frac{1}{2} \left(f_j - \frac{(2x_j + 3x_j^4)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \right) \\ a_3 &= \frac{f'_j.e^{-x_j^3}}{3.(2 + 18.x_j^3 + 9.x_j^6)} \end{aligned} \tag{2.7}$$

From equation (1.4) we have

$$\begin{aligned} y_{j+1} - y_j &= a_1(x_{j+1} - x_j) + a_2(x_{j+1}^2 - x_j^2) + a_3(e^{x_{j+1}^3} - e^{x_j^3}) \\ y'_{j+1} - y'_j &= 2a_2(x_{j+1} - x_j) + 3a_3(x_{j+1}^2 e^{x_{j+1}^3} - x_j^2 e^{x_j^3}) \end{aligned} \tag{2.8}$$

From equation(2.2) and substituting the values of a_1, a_2 and a_3 from (2.7) ,in equation (2.8) ,we have

$$y_{j+1} = y_j + h \left(\beta - x_j.f_j + \frac{(x_j^2 + 3x_j^5)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \right) + \frac{h(2x_j + h)}{2} \left(f_j - \frac{(2x_j + 3x_j^4)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \right) + \frac{(e^{3x_j h(x_j+h)+h^3} - 1)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \tag{2.9}$$

$$y'_{j+1} = y'_j + hf_j - \frac{h(2x_j + 3x_j^4)f'_j}{2 + 18.x_j^3 + 9.x_j^6} + \frac{((x_j + h)^2 e^{3x_j h(x_j+h)+h^3} - x_j^2)f'_j}{2 + 18.x_j^3 + 9.x_j^6} \tag{2.10}$$

We replace f'_j by its first order difference approximation in (2.9) i.e.

$$f'_j = \frac{f_{j+1} - f_j}{h}$$

So we will obtain our single step implicit algorithm .

Thus we have developed single step implicit algorithm of the form

$$\begin{aligned} y_{j+1} &= y_j + h.\phi(h, f_j, f'_j) \\ y'_{j+1} &= y'_j + h.\varphi(h, f_j, f'_j) \end{aligned} \tag{2.11}$$

where ϕ and φ are increment functions. These increment functions depend on h, f_j and f'_j . If the system of equations (2.10) are linear generally solved by iterative method otherwise Newton Raphson method. After application of these algorithm we have taken as an approximation for the exact solution and derivative of solution of the problem (1.1) at x_{j+1} , the values y_{j+1} and y'_{j+1} given by (2.9). Repeating the procedure along the nodes on the interval of integration, we will obtain a discrete solution and derivative of the solution for the problem. In the numerical section, we will see that the performance of proposed algorithm for a variety of second order initial value problems.

3 The Local truncation error and Convergence

In this section, we consider the error associated to the proposed algorithm (2.9). Let the local truncation error T_{n+1} , defined as in [13],

$$T_{n+1} = y(x_n + h) - y_{n+1} \quad (3.12)$$

Substituting the value of y_{n+1} from (2.9) in (3.11), and expanding $y(x_n + h)$ in Taylor series about point x_n , so we have

$$\begin{aligned} T_{n+1} &= \frac{h(x_n^2 + 3x_n^5)(y_n^{(3)} - 1)}{2 + 18x_n^3 + 9x_n^6} + O(h^2) \\ &= \frac{h}{2}(x_n^2 + 3x_n^5)(y_n^{(3)} - 1)(1 + 9x_n^3 + \frac{9}{2}x_n^6)^{-1} + O(h^2) \\ &< \frac{h}{2}(y_n^{(3)} - 1)x_n^2 + O(h^2) \\ |T_{n+1}| &< \frac{hb^2}{2}|y_n^{(3)}| \end{aligned} \quad (3.13)$$

where $b = \max(x_n)$ in $[a, b]$. Thus local truncation error T_{n+1} is bounded. We know x_0 and $y(x_0)$ exactly then using algorithm (2.9), we can compute $y_{n+1}, n = 0, 1, 2, 3, \dots, N$ with maximum error $\frac{hb^2}{2}y_n^{(3)}$. The error arising from all the initial conditions tends to zero as $h \rightarrow 0$ i.e. for large N . Similarly we can find maximum error in second algorithm of (2.9), for computation of derivative of solution. Thus we have concluded that method (2.9) is convergent for large N .

4 Stability property

To discuss stability property of the algorithm (2.9), we follow the same method as discussed in [12,13]. Consider the Dahlquist test equation for stability,

$$y''(x) = \lambda^2 y(x), \quad x \in [a, b] \quad \text{and} \quad \lambda \in R$$

subject to initial conditions $y(x_0) = y_0, y'(x_0) = \lambda y_0$. Apply the method (2.9) to this test equation, we obtained a finite difference equation, assuming the negligible contribution of the terms with $O(h^2)$ and $O(x_n^2)$ in the expression,

$$\begin{aligned} y_{n+1} &= (1 + h\lambda)y_n \\ &= e^{h\lambda}y_n \\ &\simeq E(h\lambda)y_n, \quad n = 0, 1, 2, \dots, N \end{aligned} \quad (4.14)$$

where the stability function $E(h\lambda)$ is an approximation to $e^{h\lambda}$. For the algorithm (2.9) to be converge

$$|E(h\lambda)| < 1 \quad (4.15)$$

Solving inequality (4.14), thus we obtained the corresponding interval of absolute stability of (2.9) is $(-2, 0)$.

5 Numerical experiment

In this section, four numerical examples linear and nonlinear were considered, to illustrate our algorithm (2.9) and to demonstrate computationally its efficiency and accuracy. In tables, we have shown maximum absolute error computed on the nodal points in the interval of integration for these examples in their solution and derivative of solution. Let y_i and y'_i are the numbers calculated by (2.9) respectively which are an approximate value of the theoretical solution $y(x)$ and derivative of solution i.e. $y'(x)$ at the point $x = x_i$. Maximum absolute error is calculated in both solution and derivative of solution by

$$MAE(y) = \max_i |y(x_i) - y_i|$$

$$MAE(y') = \max_i |y'(x_i) - y'_i|, \quad i = 1, 2, \dots, N.$$

All computations in the examples considered were performed in the GNU FORTRAN environment version -99 compiler (2.95 of gcc) running on a MS Window 2000 professional operating system.

Example 5.1. Consider the initial value problem,

$$y''(x) = \left(\frac{-40}{x^3} + \frac{400}{x^4} \right) y(x)$$

The exact solution in $[1, 2]$ is $y(x) = e^{\left(\frac{-20}{x}\right)}$. The maximum absolute error in $y(x)$ and $y'(x)$ are given Table 1.

Example 5.2. Consider nonlinear initial value problem

$$y''(x) = 6y^2(x)$$

The exact solution in $[0, 1]$ is $y(x) = (1+x)^{-2}$. The maximum absolute error in $y(x)$ and $y'(x)$ are given Table 2.

Example 5.3. Consider nonlinear initial value problem

$$y''(x) = y^3(x) - y(x)y'(x)$$

The exact solution in $[1, 2]$ is $y(x) = (1+x)^{-1}$. The maximum absolute error in $y(x)$ and $y'(x)$ are given in Table 3.

Example 5.4. Consider nonlinear initial value problem

$$y''(x) = y^2(x) + \frac{1}{2} \cos(x) - \sin^4\left(\frac{x}{2}\right)$$

The exact solution in $[0, 1]$ is $y(x) = \sin^2\left(\frac{x}{2}\right)$. The maximum absolute error in $y(x)$ and $y'(x)$ are given Table 4.

Table 1 Maximum absolute error in $y(x) = e^{\left(\frac{-20}{x}\right)}$ and $y'(x)$ for Example 5.1 .

MAE	N						
	64	128	256	512	1024	2048	4096
y	.5621729(-3)	.2748708(-3)	.1313720(-3)	.5965512(-4)	.2474944(-4)	.1009797(-4)	.4115111(-5)
y'	.5267575(-6)	.1289518(-6)	.3187597(-7)	.7879862(-8)	.1913576(-8)	.5345364(-9)	.1852914(-9)

Table 2 Maximum absolute error in $y(x) = (1+x)^{-2}$ and $y'(x)$ for Example 5.2 .

MAE	N						
	128	256	512	1024	2048	4096	8192
y	.9100055(-2)	.5043587(-2)	.2747672(-2)	.1475068(-2)	.7822510(-3)	.4106779(-3)	.2136840(-3)
y'	.1660321(-1)	.8981451(-2)	.4802107(-2)	.2541303(-2)	.1330271(-2)	.6887614(-3)	.3566145(-3)

Table 3 Maximum absolute error in $y(x) = (1+x)^{-1}$ and $y'(x)$ for Example 5.3 .

MAE	N						
	128	256	512	1024	2048	4096	8192
y	.1363998(-1)	.7091403(-2)	.3645122(-2)	.1860261(-2)	.9453296(-3)	.4789233(-3)	.2411603(-3)
y'	.6387859(-2)	.3583610(-2)	.1952946(-2)	.1042857(-2)	.5497336(-3)	.2868771(-3)	.1467168(-3)

Table 4 Maximum absolute error in $y(x) = \sin^2\left(\frac{x}{2}\right)$ and $y'(x)$ for Example 5.4 .

MAE	N						
	128	256	512	1024	2048	4096	8192
y	.3083482(-1)	.2080867(-1)	.1396590(-1)	.9322166(-2)	.6191766(-2)	.4095947(-2)	.2701349(-2)
y'	.2283364(-2)	.1416236(-2)	.8000433(-3)	.4311204(-3)	.2279877(-3)	.1192688(-3)	.6163130(-4)

6 Conclusion

In this paper ,we have described a new method that is efficient ,stable and convergent for solving second order initial value problems in ordinary differential equations.The implementation of the method is simple.The results we obtained for examples shows that method is computationally efficient and accurate. Our future works will deal with extension of the present method to solve higher order boundary value problems and improving its order of accuracy.

References

- [1] Lambert J. D., *Numerical Methods for Ordinary Differential systems*, John Wiley, England (1991).
- [2] Henrici P., *Discrete Variables Methods in Ordinary Differential Equation*, John Wiley and Sons New York (1962).
- [3] Cash J.R. and Wright M.H., A deferred correction method for nonlinear two-point boundary value problem, *SIAM J. Sci. Stat. Comp.* no. 12,971-989 (1991). doi: [10.1137/0912052](https://doi.org/10.1137/0912052)

- [4] Collatz L., Numerical Treatment of Differential Equations (3/e), Springer Verlag, Berlin (1966).
- [5] Mickens R.E., Nonstandard Finite Difference Models of Differential Equations, World Scientific, Singapore (1994).
- [6] Sunday J. and Odekunle M.R., A New Numerical Integrator for the Solution of Initial Value Problems in Ordinary Differential Equations, The Pacific Journal of Science and Technology, Vol.-13, no. 1, pp. 221-227 (2012).
- [7] Fatunla S.O., A New Algorithm for Numerical Solution of Ordinary Differential Equations, Computer and Mathematics with Applications, no. 2, 247-253 (1973).
- [8] Keller H. B., Numerical Methods for Two Point Boundary Value Problems, Blaisdell Waltham Mass. (1968).
- [9] Stoer J. and Bulirsch R., Introduction to Numerical Analysis (2/e), Springer-Verlag, Berlin Heidelberg (1991).
- [10] Baxley J.V., Nonlinear Two Point Boundary Value Problems in Ordinary and Partial Differential Equations (Everitt, W.N. and Sleeman, B.D. Eds.), 46-54, Springer-Verlag, New York (1981).
- [11] Gear C.W., [Numerical Initial Value Problems in Ordinary Differential Equations](#), Prentice Hall (1971).
- [12] Dahlquist G., [A special stability problem for linear multistep methods](#), BIT, 3, 27-43 (1963).
- [13] Jain M. K., Iyenger S. R. K. and Jain R.K., Numerical Methods for Scientific and Engineering Computations (2/e), Wiley Eastern Ltd. New Delhi (1987).

This page is intentionally left blank