Nigerian Journal of Mathematics and Applications
Volume A, **28**, (2019), 1 − 17. *Printed by Unilorin press*
©*Nig. J. Math. Appl.* *http://www.njmaman.com*

Efficient *k*-Derivative Methods for Lane-Emden Equations and Related Stiff Problems

M. O. Ogunniran[1*], Y. Haruna[2] and R. B. Adeniyi[3]

Abstract

Lane-Emden equations are one of singular classes of problems in mathematical physics. Existing methods for this class of problems are series and analytical methods. This study focuses on the exploration of the possibility of a class of multi-derivative methods for approximating its solution. Taylor's series expansion was applied on a new multi-derivative backward differentiation formula to obtain a new class of k-step, k-derivative method containing one super-future point evaluation. The k-derivative methods were applied to singular Lane-Emden Equations and related stiff equations. Matlab codes were written for the solution of problems considered and the results obtained showed the competitive strength of these methods with other methods. The methods developed have good stability properties, a high degree of accuracy and are effective.

1. Introduction

Some problems in the literature of mathematical physics can be formulated as equation of Lane-Emden type. The Lane-Emden equation is a second order

Received: 11/01/2019, Accepted: 22/03/2019, Revised: 30/03/2019. * Corresponding author.
2015 *Mathematics Subject Classification.* 46G05, & 32Q26.
Key words and phrases. k-derivative, Super-future point, Backward Differential Formula, Singular Lane-Emden Equations, Stiff Equations, Stability Properties, A-stability
[1]Department of Mathematical Sciences, Osun State University Osogbo
[2]Department of Mathematics, Saadatu Rimi College of Education Kumbotso, Kano State
[3]Department of Mathematics, University of Ilorin, Ilorin Nigeria
E-mail: muideen.ogunniran@uniosun.edu.ng

Ordinary Differential Equation (ODE) which is singular at $x = 0$. The general Lane-Emden type equations are formulated as:

$$
\begin{aligned}
&y''(x) + \tfrac{\alpha}{x}y(x) + f(x)g(y) = h(x),\ \alpha \geq 0, \\
&y(0) = A,\ y'(0) = B.
\end{aligned}
\tag{1}
$$

where α, A, and B are real constants and $f(x)$, $g(x)$, and, $h(x)$ are some given functions.
The peculiarity of Lane-Emden equation at $x = 0$ causes a stiff scenario at the start of integration.

2 Literature Review

Lately, the problem of obtaining efficient methods for the solution of stiff systems has been the focus for researchers and as a result, much successful class of methods have been proposed. Enright (1974) proposed a second derivative method for the solution of stiff system of ODEs and the stability properties of the A-stable, high order method was improved by Cash (1981), Ismail and Ibrahim (1999), Hojjati, Rahimi-Ardabili and Hosseini (2006) and Mehdizadeh, Rahimi-Ardabili, and Hojjati (2009). In order to improve on these results, many other methods have been proposed. Dahlquist (1963) showed that A-stability linear multi-step method, for the solution of stiff problems or related systems, has to be of maximum order 2 and must be implicit. According to Hairer and Wanner (1996), the search for higher order A-stable multi-step methods can be carried out in two main ways: using higher derivatives of solutions and including some additional stages, such as off-step points or super-future points or the like of it.
Butcher (1966) introduced the general linear methods as a unique methodology for the traditional methods to study the properties of consistency, stability and convergence, and to formulate new methods with a clear advantage over traditional methods. According to Hojjati (2015), one of the main directions to construct methods with higher order and extensively stability region, is using higher derivatives of the solutions and some methods have been introduced that have good properties, especially for stiff problems as in (Enright (1974), Cash (1981), and Hojjati, Ardabili & Hosseini (2006)). Although general linear methods include traditional first derivative methods, they were extended to second derivative general linear methods to cover second derivatives. Second derivatives method was presented by Cash (1981) and a modified method was also presented in Cash (1983). Cash (2000) applied his modified method to the solution of stiff IVPs and Differential-Algebraic Equations (DAEs). New classes of methods were introduced by Butcher and Hojjati (2005) and were studied more in (Abdi & Hojjati (2011(a) & 2011(b)), Abdi, Bras & Hojjati (2014), and Ezzeddine, Hojjati & Abdi (2014)). A special class of this method was presented by Ismail & Ibrahim (1999) and Mehdizadeh, Oskuyi & Hojjati (2006, 2012). The class presented by

Hojjati *et al.* (2006) was explored in the solution of singular Lane-Emden type of equation in Hojjati & Parand (2011).

In the quest of exploration of super-future points technique of approximation, Most works in this class of linear multi-step methods have been based on Backward Differentiation Formula (BDF), because of its special properties (Hojjati, 2015). Different authors have produced different modifications and among the first is the Extended Backward Differentiation Formulae (EBDF) by Cash (1980) in which one super-future point technique was applied. This method was A-stable up to order 4 and A(α)-stable up to order 9. Optimization of necessary computations of EBDF was carried out by Cash (1983) and Hosseini & Hojjati (1999) using a Modified Extended Backward Differentiation Formulae (MEBDF) and Matrix Free MEBDF (MF-MEBDF) respectively. Using one free parameter technique and with a blended application of implicit and explicit extended backward differentiation formulae, the work of Fredbeul (1988) and Hojjati, Rahimi and Hosseini (2004) were introduced. Ebadi (2011) proposed a class of multi-step methods for Solving Initial Value Problems (IVPs). The method was based on super-future points technique. An extended hybrid BDF Methods was developed by Ebadi & Gokhale (2014) and was applied to nonlinear parabolic PDEs. A class of parallel methods with a super-future points technique for the numerical solution of stiff systems was proposed by Hojjati (2015). Kleefeld & Martin-Vaquro (2016) in their paper, SERK2v3: Mildly Stiff Nonlinear PDEs, presented the solution of parabolic PDEs. Some continuous second derivative multi-step methods for numerical integration of IVP in ODEs was presented in the work of Awoshiyan (2017) and Jator & Coleman (2017) presented a nonlinear second derivative method with a variable step-size based on continued fractions for singular IVPs and applied same to the solution of some parabolic PDEs.

It is on this note we have the motivation to obtain a class of high derivative methods with a high order and A-stability property which are consistent and convergent using one super-future technique over two parameters. This is for the purpose of improving the accuracy and stability of the class of methods then use for solution of singular and stiff problems in ODEs.

3 Development and Analysis of Methods

This section contains the developmental process of the proposed methods.

3.1 Development.

Multi-derivative Multi-step Methods. The general linear multi-derivative multi-step method is given as:

$$\sum_{j=0}^{k} \alpha_j y_{n+j} + \sum_{i=0}^{k} \sum_{j=0}^{l} h^j \alpha_{ij} f_{n+i}^{(j-1)} = 0, \quad n = 0, 1, 2, ... \tag{2}$$

for solving IVP: $y' = f(x,y),\ y(0) = y_0$, where y_n is an approximation to $y(x_n)$, $x_n = nh,\ h > 0$ and $f_m^{(j)} = f^{(j)}(x_m, y_m)$

New k-derivative k-step Methods. We propose a new class of Multi-derivative Backward Differentiation Formula of the form:

$$\sum_{i=0}^{k} \alpha_i y_{n+i} = \sum_{j=1}^{k} h^j(\beta_{jk} f_{n+k}^{(j-1)} - \beta_{jk}^{*} f_{n+k+1}^{(j-1)}); \tag{3}$$

for solving IVP: $y' = f(x,y),\ y(0) = y_0$, where y_n is an approximation to $y(x_n)$, $x_n = nh,\ h > 0$ and $f_m^{(j)} = f^{(j)}(x_m, y_m)$ such that:

$$f^{(0)}(x,y) = f(x,y),$$

$$f^{(j)}(x,y) = \frac{\partial f^{(j-1)}(x,y)}{\partial x} + f(x,y)\frac{\partial f^{(j-1)}(x,y)}{\partial y},$$

where $\alpha_k = 1,\ k \geq 1,\ \alpha_i,\ \beta_{jk}$ and $\beta_{jk}^{*},\ i = 0(1)k,\ j = 1(1)k$ are parameters to be determined. It could be observed that there are $3k$ unknown parameters $\forall\ k$. Using Taylor's series expansion, a system of $3k$ algebraic system of equations was obtained $\forall\ k$. The resulting equations were then solved to obtain these parameters. The coefficients of the k-step, k-derivative methods of class (3) are given in Table 1, for $k = 1, 2, 3, 4$. In this form we use the one super-future point technique. Assuming the values of y_{n+j} at the points $x_{n+j},\ 0 \leq j \leq k-1$ are known, we carry out the following computations:

Algorithm. Stage 1 (First Predictor): Compute y_{n+1} as the solution of:

$$y_{n+1} = y_n + hf_n \tag{4}$$

Stage 2 (Next predictor): Compute y_{n+s+1} as the solution of:

$$y_{n+s+1} = y_{n+s} + \frac{h}{2}(f_{n+s+1} - f_{n+s}), \quad s = 1(1)n+1 \tag{5}$$

Stage 3: Evaluate

$$f_{n+k}^{(j)} = f_{n+k}^{(j)}(x_{n+k}, y_{n+k})$$

$$f_{n+k+1}^{(j)} = f_{n+k+1}^{(j)}(x_{n+k+1}, y_{n+k+1}),\ \ j = 0, 1, \cdots, k-1.$$

Stage 4: Compute y_{n+k} as the solution of:

$$\sum_{i=0}^{k} \alpha_i y_{n+i} = \sum_{j=1}^{k} h^j(\beta_{jk} f_{n+k}^{(j-1)} - \beta_{jk}^{*} f_{n+k+1}^{(j-1)}); \tag{6}$$

where the coefficients of the k-step, k-derivative class of method (3) are given in Table 1, for $k = 1, 2, 3, 4$.

Table 1: Coefficients in (3)

k	1	2	3	4
α_0	-1	$\frac{31}{481}$	$\frac{-211436401}{41382290065}$	$\frac{-1656959992994038067505}{1500469244653262037100097}$
α_1	1	$\frac{-512}{481}$	$\frac{3099770127}{41382290065}$	$\frac{2368014282706371441664 0}{1500469244653262037100097}$
α_2		1	$\frac{-44270623791}{41382290065}$	$\frac{-1711774453308930872671 68}{1500469244653262037100097}$
α_3			1	$\frac{-892661968502792792064 1}{1500469244653262037100097}$
α_4				1
j		$(\beta_{jk}, \beta^*_{jk})$		
1	$(\frac{3}{2}, \frac{1}{2})$	$(\frac{178}{481}, \frac{-272}{481})$	$(\frac{48111979602}{41382290065}, \frac{9406586862}{41382290065})$	$(\frac{1828077040052712492448 7556}{1500469244653262037100097}, \frac{1800193543646605094814 1056}{1500469244653262037100097})$
2		$(\frac{-374}{481}, \frac{92}{481})$	$(\frac{-9633260682}{41382290065}, \frac{-2152612098}{41382290065})$	$(\frac{8190683849580422766704 568}{1500469244653262037100097}, \frac{-9561755328017425020840 960}{1500469244653262037100097})$
3			$(\frac{6747790932}{41382290065}, \frac{235332}{41382290065})$	$(\frac{1580241991914841688056 800}{1500469244653262037100097}, \frac{2001530633726483628493 824}{1500469244653262037100097})$
4				$(\frac{436107575634834902 4}{1500469244653262037100097}, \frac{-1690619323664772166394 88}{1500469244653262037100097})$

Analysis of Methods. This section contains the analysis of the proposed methods.

3.2 Analysis.

3.2.1 Order and Error Constant. In this section, we analyze the methods using Taylor's series method for derivation of order and error constant of the methods. We associate the linear difference operator $\mathcal{L}$ defined by:

$$(7)\quad \mathcal{L}[y(x);h] = \sum_{i=0}^{k} \alpha_i y(x+ih) - \sum_{j=1}^{k} h^j (\beta_{jk} y^{(j)}(x+ih) - \beta^*_{jk} y^{(j)}(x+(i+1)h))$$

where $y(x)$ is an arbitrary function, continuously differentiable on $[a,b]$.
Expanding $y(x+ih)$ and its derivatives $y^{(j)}(x+ih)$ as Taylor's series about x, and collecting terms gives:

$$(8)\qquad \mathcal{L}[y(x);h] = c_0 y(x) + c_1 h y^{(1)} y(x) + \cdots + c_q h^q y^{(q)}(x) + \cdots$$

where c_q are constants.
The class of method (3) is of order p if:

$$(9)\qquad \mathcal{L}[y(x);h] = c_{p+1} h^{p+1} y^{(p+1)}(x) + o(h^{p+2}); \quad c_{p+1} \neq 0$$

where

$$\left.\begin{aligned} c_0 &= \alpha_0 + \alpha_1 + \alpha_2 + \cdots + \alpha_k, \quad \alpha_k = 1 \\ c_1 &= \alpha_1 + 2\alpha_2 + \cdots + k\alpha_k - (\beta_{1k} - \beta_{1k}^*) \\ &\vdots \\ c_q &= \tfrac{1}{q!}(\alpha_1 + 2^q\alpha_2 + \cdots + k^q\alpha_k) - \textstyle\sum_{j=1}^{k} \tfrac{(k)^{q-j}}{(q-j)!}\beta_{jk} + \textstyle\sum_{j=1}^{k} \tfrac{(k+1)^{q-j}}{(q-j)!}\beta_{jk}^* \end{aligned}\right\} \tag{10}$$

Also, if we define the polynomial;

$$\rho_j(\xi) = \sum_{i=0}^{k} \alpha_i \xi^i \tag{11}$$

we shall always assume that the polynomial has no common factors, then:

$$\frac{c_{p+1}}{\rho_0'(0)} \tag{12}$$

is the Error constant.
The table below shows the Order and Error Constant of class of methods (3) for $k = 1, 2, 3, 4$.

Table 2: Order and Error Constant of (3)

k	Order	Error Constant
1	2	$+\frac{5}{12}$
2	5	$-\frac{5}{666}$
3	8	$-\frac{318329027}{2417408243640}$
4	11	$-\frac{786576141489554093766656}{412629042279647060202766675}$

3.2.2 Consistency and Zero-stability. Definition (Lambert, 1973): The linear multi-step method (3) is said to be consistent if it has order $p \geq 1$.
Definition (Lambert, 1973): The linear multi-step method (3) is said to be zero-stable if no root of the first characteristic polynomial $\rho(\xi)$ has modulus greater than one, and if every root with modulus one is simple.
Thus, we now introduce the first and second characteristic polynomials of the method as defined as:

$$\begin{aligned} \rho(\xi) &= \textstyle\sum_{j=0}^{k} \alpha_j \xi^j \\ \sigma(\xi) &= (\beta_{1k} - \beta_{1k}^*)\xi^j \end{aligned} \tag{13}$$

It follows from (13), that:

$$\rho(1) = 0; \quad \rho'(1) = \sigma(1) \tag{14}$$

Thus, for a consistent method, the first characteristics polynomial always has a root at ± 1.

3.2.3 Convergence. Theorem (Lambert, 1973): *The necessary and sufficient conditions for linear multi-step method to be convergent are that it be consistent and zero-stable.* Thus, since the method (3) is consistent and zero-stable; we say it is convergent.

3.2.4 Stability. We investigate the stability properties of the method by applying it on the test IVP:

(15) $$y' = \lambda y,\ y'_n = \lambda y_n,\ y''_n = \lambda^2 y_n, \cdots, y_n^{(n)} = \lambda^n y_n.$$

Putting (15) in (3), we get the characteristic equation:

(16) $$\sum_{j=0}^{k} \alpha_j \xi^j - \sum_{j=1}^{k} z^j (\beta_{jk} \xi^k - \beta_{jk}^* \xi^{k+1}) = 0, \quad z = h\lambda$$

Stability Domain of the Methods

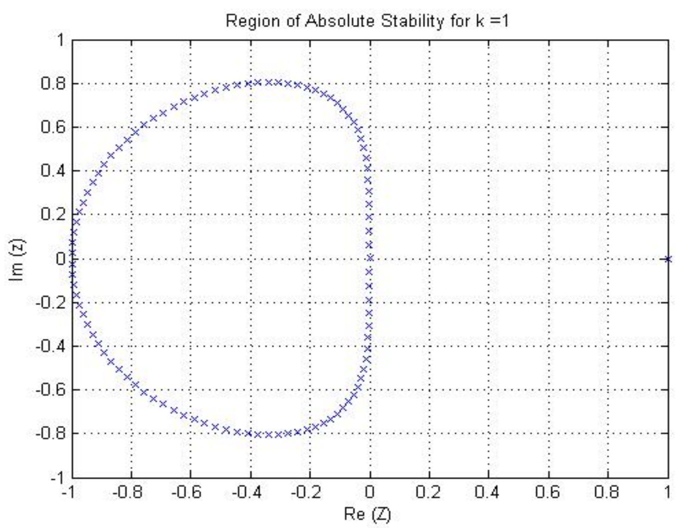

Figure 1: Stability Domain for k=1

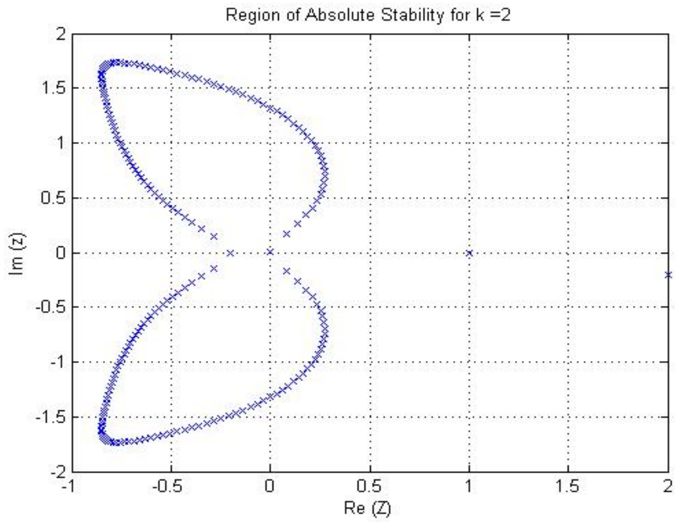

Figure 2: Stability Domain for k=2

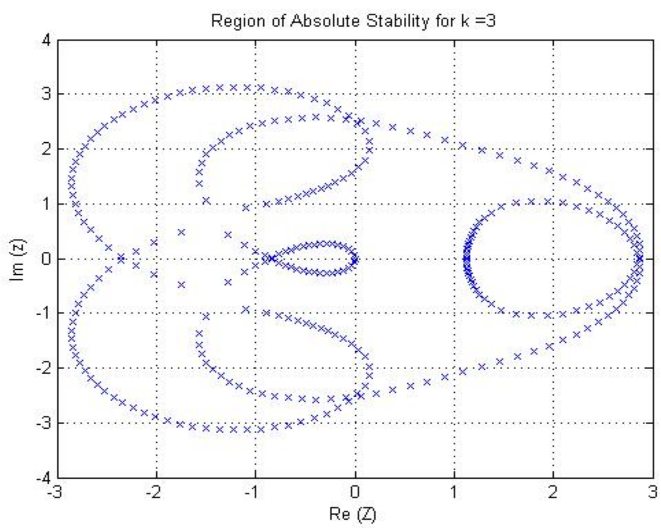

Figure 3: Stability Domain for k=3

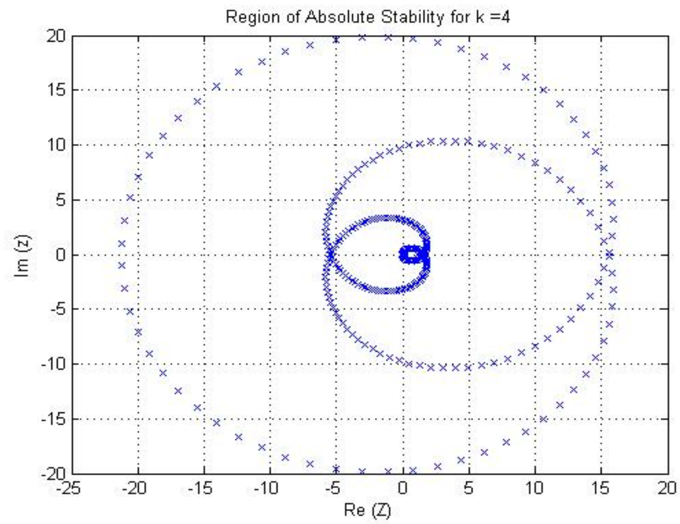

Figure 4: Stability Domain for k=4

4 Computational Results

In this section, the new methods are implemented on the solution of Lane-Emden equations and related stiff problems.

4.1 Numerical Example

Example 1. Consider the nonlinear system of stiff differential equations of the form (Mehdizadeh *et al.*, 2012):

$$\begin{gathered} y_1' = \lambda y_1 + y_2^2, \;\; y_1(0) = \tfrac{-1}{\lambda+2}, \\ y_2' = -y_2, \;\; y_2(0) = 1, \;\; \lambda = 10000 \\ y_1(x) = \tfrac{-exp(-2x)}{\lambda+2}, \quad y_2(x) = \exp(-x). \end{gathered} \tag{17}$$

See Table 3 for the results.

Example 2. Consider the following stiff equation of the form (Mehdizadeh *et al.*, 2012):

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} -\alpha & -\beta \\ \beta & -\alpha \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} (\alpha+\beta-1)\exp(-x) \\ -(\alpha-\beta-1)\exp(-x) \end{pmatrix}$$

$$(y_1(0), y_2(0))^T = (1,1)^T \tag{18}$$

The required solution is:

$$y_1(x) = y_2(x) = \exp(-x)$$

See Table 4 for numerical results.

Example 3. Consider the Lane-Emden equation of the form (Hojjati & Parand, 2011):

$$\begin{gathered} y''(x) + \frac{2}{x}y'(x) - 2(2x^2+3)y = 0, \quad x \geq 0 \\ y(0) = 1, \ y'(0) = 0 \\ y(x) = \exp(x^2). \end{gathered} \tag{19}$$

See Table 5 for numerical results.

Example 4. Considering the two-point boundary value problem of Lane-Emden type (Ascher & Petzold, 1997; pg 188):

$$\begin{gathered} y''(x) + \frac{4}{x}y'(x) + (xy(x) - 1)y(x) = 0, \quad 0 < x < \infty \\ y'(0) = y(\infty) = 0. \end{gathered} \tag{20}$$

This is a well-behaved problem with a smooth, nontrivial solution. To solve it numerically, the interval $[0, \infty]$ by a finite, large interval $[0, L]$ and require:

$$y(L) = 0. \tag{21}$$

See Table 3 for computational results.

4.2 Discussion of Results

Table 3: Numerical Comparison for Example 1 for $h = 0.0001$

n	x	y_i	Exact	Error of Mehdizadeh *et al.* (2012)	Error of New Method
	3	y_1	-2.4782565254 E -07	2.478147 E -11	2.571397 E -15
		y_2	4.9787068368 E -02	2.471093 E -06	2.109007 E -13
	5	y_1	-4.5390851592 E -09	3.450271 E -14	4.709278 E -17
		y_2	0.0067379470 E -03	2.304573 E -08	2.854314 E -14
	10	y_1	-2.0607414741 E -13	3.456372 E -18	2.153933 E -21
		y_2	4.5399929762 E -05	3.150734 E -10	1.695692 E -16
CPU time			NA	NA	6.03125s

It could be observed from Table 3 that the new method shows superiority over the method of Mehdizadeh *et al.* (2012) interms of accuracy. The computation time of new method was also presented. The new method executes the problem in 6.03125s.

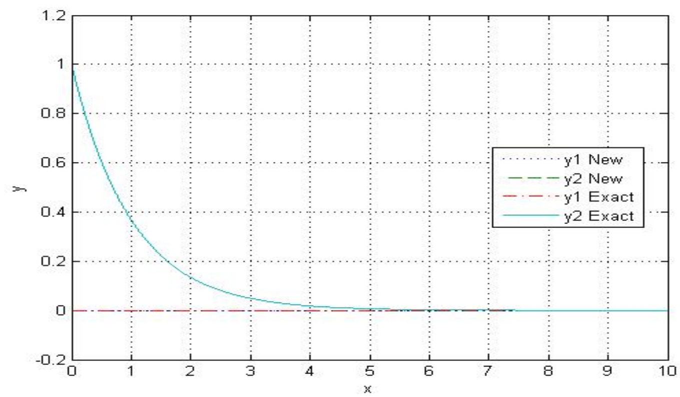

Figure 5: Graphical Comparison of New Method with Exact for Example 1

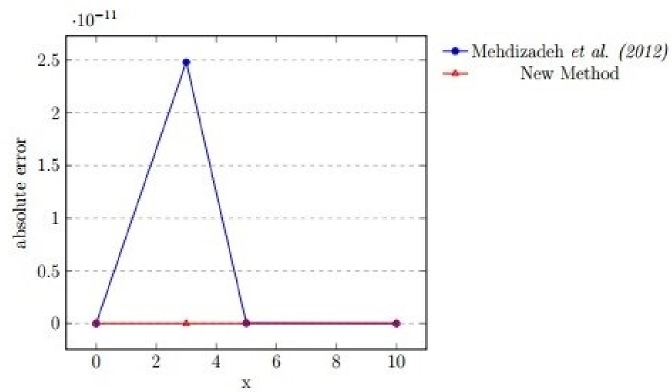

Figure 6: Comparison Graph of Absolute Error for $y_1(x)$ in Example 1

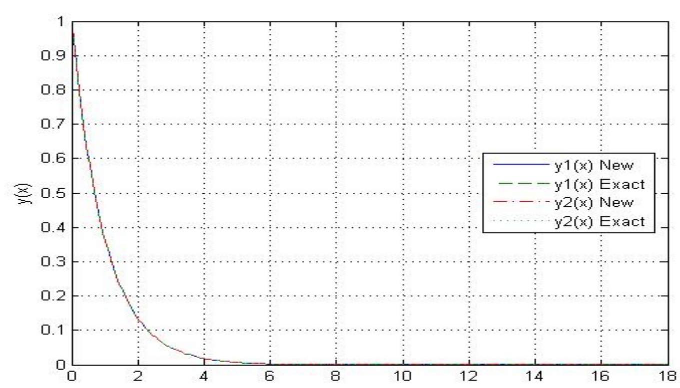

Figure 8: Comparison Graph of Absolute Error for $y_2(x)$ in Example 1

Table 4: Numerical Comparison for Example 2 for $\alpha = 1,\ \beta = 30,\ h = 0.09$

x	y_i	Exact	Error of E2BD Cash (1981) 1	Error of E2BD Cash (1981) 2	Mehdizadeh *et al.* (2012)	New Method (3)
4.5	y_1	1.110899654 E -02	<1.000000 E -11	<1.000000 E -11	3.000000 E -12	8.482352 E -12
	y_2	1.110899654 E -02	<1.000000 E -11	<1.000000 E -11	3.000000 E -12	3.551161 E -12
9.0	y_1	1.234098041 E -04	<1.000000 E -13	<1.000000 E -13	3.000000 E -15	9.212352 E -14
	y_2	1.234098041 E -04	<1.000000 E -13	<1.000000 E -13	3.000000 E -15	3.856778 E -14
13.5	y_1	1.370959086 E -06	<1.000000 E -16	1.000000 E -12	7.000000 E -17	1.000517 E -15
	y_2	1.370959086 E -06	<1.000000 E -16	1.000000 E -12	6.000000 E -17	4.188690 E -16
18.0	y_1	1.522997974 E -08	<1.000000 E -18	1.000000 E -12	1.000000 E -20	3.384886 E -13
	y_2	1.522997974 E -08	<1.000000 E -18	1.000000 E -12	2.000000 E -20	1.136459 E -12
CPU time			NA	NA	NA	0.265625s

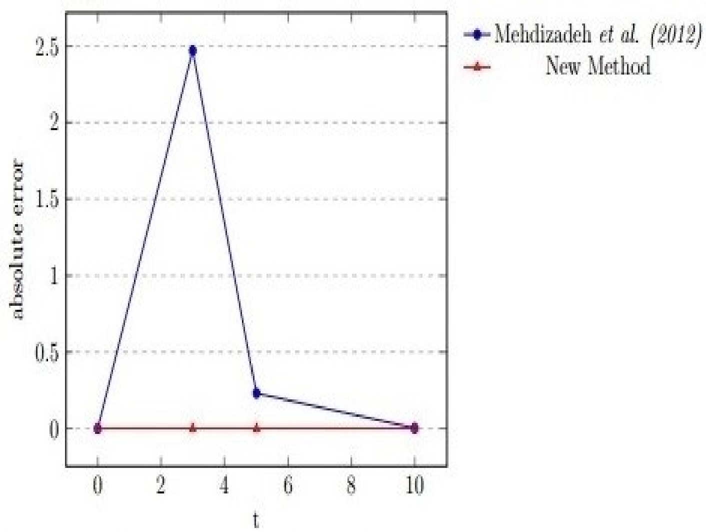

Figure 7: Graphical Comparison of New Method with Exact for Example 2

Remark

The formula of E2BD-Class 1 and Class 2 reported by Cash (1981) with order 7 is unstable for this problem. The formula reported in Mehdizadeh $etal$. (2012) is of order 7 and when compared with the new formula of order, 5 for large x, the new formula gives relatively close results, where error is absolute.

Table 5: Numerical Comparison for Example 3 for $h = 0.001$

n	x	Exact	Error of Hojjati & Parand (2011)	Error of New Method
	0.25	1.0644944589	1.770000 E -13	0.000000 E 00
	0.50	1.2840254167	2.140000 E -13	2.220446 E -16
	0.75	1.7550546570	2.930000 E -13	2.220446 E -16
	1.00	2.7182818285	4.540000 E -13	4.553855 E -09
CPU time		NA	NA	0.203125s

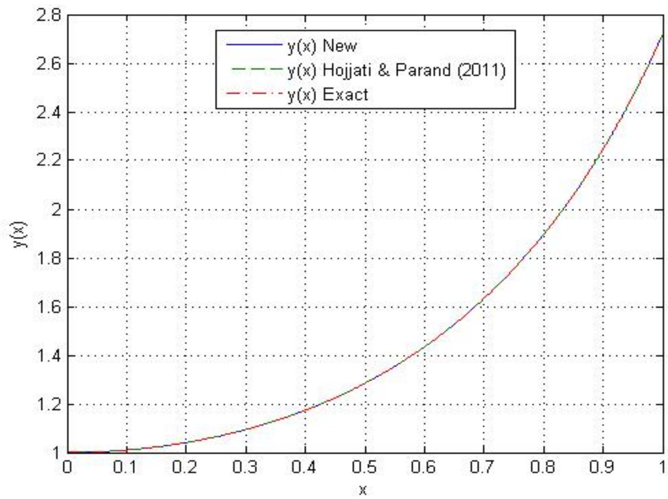

Figure 9: Graphical Comparison of New Method with Exact for Example 3

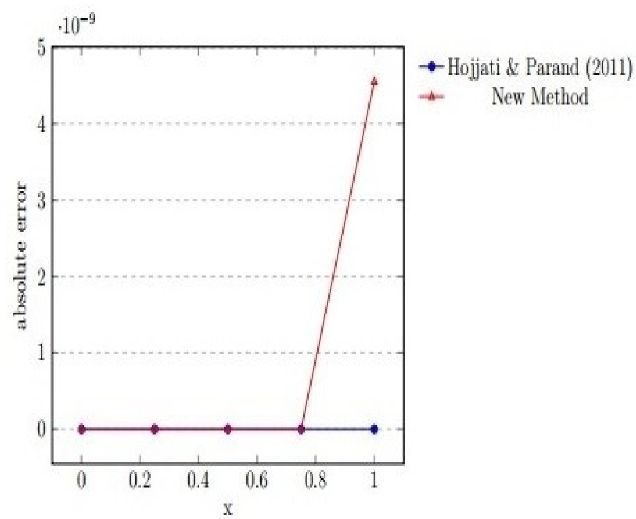

Figure 10: Comparison Graph of Absolute Error for $y(x)$ in Example 3

The new method shows superiority over the method of Hojjati & Parand (2011) in the singular problem of Lane-Emden type equation considered.

Error is absolute error obtained from the absolute difference between method and exact

Table 6: Numerical Comparison for Example 4 for $L = 20,\ h = 0.01$

n	x	Exact	Trapezoidal Method	New Method (3), $k = 1$	New Method (3), $k = 2$	ErrorT	ErrorNM1	ErrorNM2
0	0.00	1.0000000000	0.9895790336	1.0000000000	1.0000000000	1.042087 E -01	0.000000	0.00000
10	0.10	0.9991903280	0.9994740630	0.9985787057	0.9991895180	2.839649 E -04	6.121179 E -04	8.108565 E -07
20	0.20	0.9963965082	0.9966831113	0.9957552830	0.9963929085	2.876396 E -04	6.435388 E -04	3.612714 E -06
30	0.30	0.9916252651	0.9918960383	0.9910039115	0.9916169193	2.730600 E -04	6.266012 E -04	8.416284 E -06
40	0.40	0.9849050878	0.9851541327	0.9843146903	0.9848900963	2.528618 E -04	5.994461 E -04	1.522126 E -05
50	0.50	0.9762759469	0.9764999453	0.9757190939	0.9762524893	2.294418 E -04	5.653874 E -04	2.402763 E -05
60	0.60	0.9657888987	0.9659855667	0.9652650355	0.9657552550	2.036346 E -04	5.424200 E -04	3.483546 E -05
70	0.70	0.9535055817	0.9536732477	0.9530126260	0.9534601524	1.747421 E -04	5.169930 E -04	4.764450 E -05
80	0.80	0.9394976138	0.9396350629	0.9390326173	0.9394389376	1.463006 E -04	4.949417 E -04	6.245487 E -05
90	0.90	0.9238458964	0.9239522992	0.9234053868	0.9237726666	1.151738 E -04	4.768215 E -04	7.926625 E -05
100	1.00	0.9066398373	0.9067147093	0.9062200312	0.9065509152	8.258186 E -05	4.630351 E -04	9.807875 E -05
500	5.00	0.0829091375	0.0827111730	0.0826118473	0.0827027952	2.387728 E -03	3.585735 E -03	2.488776 E -03
1000	10.0	4.6316606012 E -05	4.5840471433 E -05	4.5728064580 E -05	4.5856321283 E -05	1.028000 E -02	1.270692 E -02	9.937790 E -03
1500	15.0	1.7434064512 E -10	1.7025585987 E -10	1.6956708532 E -10	1.7046401900 E -10	2.342991 E -02	2.738065 E -02	2.223593 E -02
2000	20.0	4.4216886644 E -18	4.2253253158 E -18	3.9773892727 E -18	4.2483542553 E -18	4.440913 E -02	1.004818 E -01	3.920095 E -02
CPU time			$0.2188s$	$0.2188s$	$0.2344s$			

From Table 6, the new methods compared favourably with an existing method.
The error considered is relative and calculated using |Exact - Numerical method|/Exact.
The CPU time was also computed.

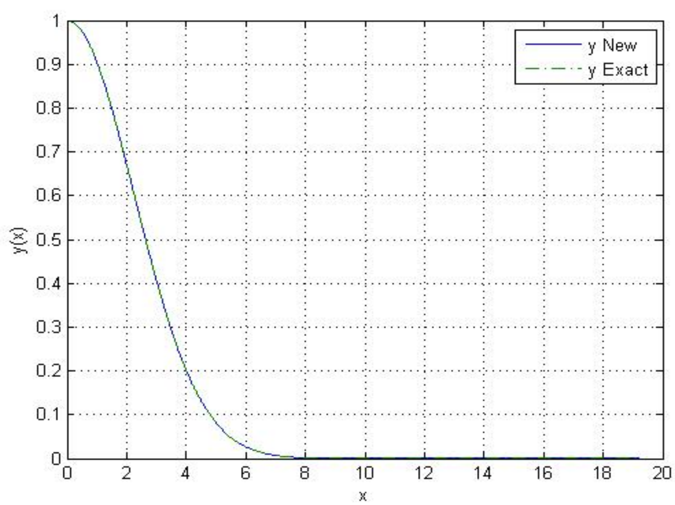

Figure 11: Graphical Comparison of New Method with Exact for Example 4

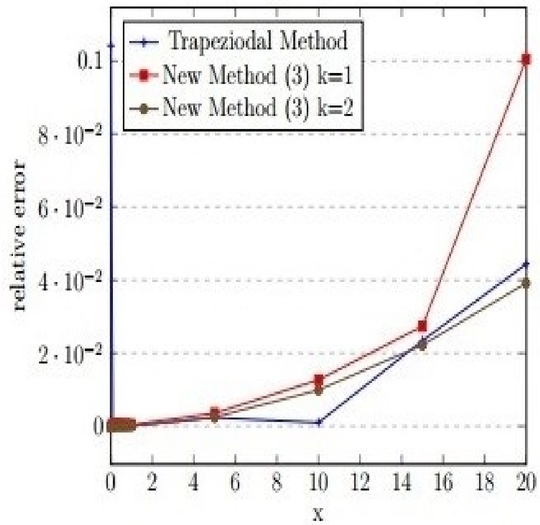

Figure 12: Comparison Graph of Relative Error for $y_1(x)$ in Example 4

Computational Details

It could be noted that at each stages 1,2 and 4 of the algorithm, a system of nonlinear equations must be solved in order to obtain the desired approximation. To solve these nonlinear systems, a Newton-Krylov solver i.e nsoli.m was used and then a direct method is used to solve any resulting system of linear equations. It is important to point that the numerical methods were programmed via MATLAB 9.2 version on a personal computer with the following specifications:

System name- Acer Aspire E15
Processor- Intel(R) Pentium(R) CPU N3530 @ 2.16GHz
Installed memory (RAM)- 4.00GB
System Type- 64-bits Operating System, x64-based processor
Operating system- 3.9 Windows Experience Index

Moreso, computational experiments were done with software optimization and only the points of emphases were shown.

Conclusion

It could be concluded, from Figures 1-4, that the new method which is a $k-$derivative method with one super point is A-stable for order p, $p = 3$ and A(α)-stable for order p, $p = 5, 8, 11$.

The table below shows the nature and interval of stability for $k = 1, 2, 3, 4$.

Table 7: Nature and Interval of Stability for new Methods

Method	Nature of Stability	Interval of Stability
$k = 1$	A-stable	[1,0]
$k = 2$	A(α)-stable	(-3,3)
$k = 3$	A(α)-stable	$(-1, \frac{1}{4}]$
$k = 4$	A(α)-stable	[-21,16]

On the other hand, Tables 3-6 demonstrate that the new method, for which the MATLAB code nsoli.m was used, has great execution in sparing CPU time when compared to the methods of Cash (1981), Hojjati & Parand (2011) and Mehdizadeh *et al.* (2012). Also, from Figures 5 - 12, the relative and absolute errors of the new methods introduced demonstrate the new methods created more precise outcomes when compared with these existing methods with the least time of computations.

References

[1] Abdi A. and Hojjati G. (2011a). An Extension of General Linear Methods. Numer. Algorithms. **57**(12), 149-167.

[2] Abdi A. and Hojjati G. (2011b). Maximal Order for Second Derivative General Linear Methods with Runge-Kutta Stability. Appl. Numer. Math., **61** (10), 1046-1058.

[3] Abdi A., Bras M. and Hojjati G. (2014). On the Construction of Second Derivative Diagonally Implicit Multi-stage Integration Methods for ODEs. Appl. Numer. Math. **76**, 1-18.

[4] Awoshiyan Y. O. (2017). Some Continuous Second Derivative Multi-step Methods for Numerical Integration of IVP in ODEs. Unpublished Doctoral Thesis, University of Ilorin, Ilorin, Nigeria.

[5] Ascher U. M. and Petzold L. R. (1997). Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations, SIAM.

[6] Butcher J. C. (1966). On the Convergence of Numerical Solutions to ODEs. Math. Comp. **20**, 1-10.

[7] Butcher J. C. and Hojjati G. (2005). Second Derivative Methods with RK Stability. Numer. Algorithms. **40** (4), 415-429.

[8] Cash J. R. (1980). On the Integration of Stiff Systems of ODEs using Extended Backward Differentiation Formulae. Numer. Math. **34** (3), 235-246.

[9] Cash J. R. (1981). Second Derivative Extended Backward Differentiation Formula for the Numerical Integration of Stiff Systems. SIAM J. Numer. Anal. **18**, 21-36.

[10] Cash J. R. (1983). The Integration of Stiff Initial Value Problems in ODEs using Modified Extended Backward Differentiation Formulae. Comp. Mathematics with Applications. **9** (5), 645-657.

[11] Cash J. R. (2000). Modified Extended Backward Differentiation Formulae for the Numerical Solution of Stiff Initial Value Problems in ODEs and DAEs. Journal of Computational and Applied Mathematics. **125**, 117-130.

[12] Kelley C. T. (2003). Solving Nonlinear Equations with Newton's Method, Fundamentals of Algorithms, SIAM, Philadelphia.

[13] Dahlquist G. (1963). A Special Stability problem for Linear Multi-step Methods. **3**, 27-43.

[14] Ebadi M. (2011). A Class of Multi-step Methods based on Super-future Points Technique for Solving IVPs. Computers and Mathematics with Applications. **61** (11), 3288-3297.
[15] Ebadi M. and Gokhale (2014). Solving Nonlinear Parabolic PDEs via Extended Hybrid BDF Methods. Indian J.Pure Appl. Math. **45** (3), 395-412.
[16] Enright W. H. (1974). Second Derivative Multi-step Methods for Stiff ODEs. SIAM J. Numer. Anal., **11**, 321-331.
[17] Ezzeddine A. K., Hojjati G. and Abdi A. (2014). Sequential Second Derivative General Linear Methods for Stiff Systems. Bulletin of the Iranian Mathematical Society. **40** (1), 83-100.
[18] Fredebeul C. (1998). A-BDF: A Generalization of the Backward Differentiation Formulae. SIAM Journal on Numerical Analysis. **35** (5), 1917-1938.
[19] Hairer E. and Wanner G. (1996). Solving Ordinary Differential Equation II: Stiff and Differential-Algebraic Problems. Springer, Berlin.
[20] Hojjati G. Rahimi M. Y. and Hosseini S. M. (2004). A-EBDF: An Adaptive Method for Numerical Solution of Stiff Systems of ODEs. Mathematics and Computers in Simulation. **66** (1), 33-41.
[21] Hojjati G., Rahimi-Ardabili M. Y., and Hosseini S. M. (2006). New Second Derivative Multi-step Methods for Stiff System. Applied Mathematical Modelling. **30**, 466-476.
[22] Hojjati G. and Parand K. (2011). An Efficient Computational Algorithm for Solving the Nonlinear Lane-Emden type Equations. International Journal of Mathematical, Computation, Physical, Electrical and Computer Engineering. **5 8**, 889-894.
[23] Hojjati G. (2015). A Class of Parallel Methods with Super-future Points Technique for the Numerical Solution of Stiff Systems. Journal of ordern Methods in Numerical Mathematics. **6** (2), 57-63.
[24] Hosseini S. M. and Hojjati G. (1999). Matrix Free MEBDF Method for the Solution of Stiff Systems of ODES. Mathematical and Computer Modelling. **29**, 67-77.
[25] Ismail G. and Ibrahim I. (1999). New Second Derivative Multi-step Methods for Stiff System. Applied Mathematical Modeling. **23**, 279-288.
[26] Jator S. N. and Coleman N. (2017). A Nonlinear Second Derivative Method with a Variable Step-size Based on Continued Fractions for Singular Initial Value Problems. Cogent Mathematics. **65** (1), 1-13.
[27] Mehdizadeh K. M., Rahimi-Ardabili M. Y. and Hojjati, G. (2009). The New Class of Super-implicit Second Derivative Multi-step Methods for Stiff Systems. Journal of Applied Functional Analysis. **4** (3), 492-500.
[28] Mehdizadeh K. M., Oskuyi N. N. and Hojjati G. (2012). A Class of Second Derivative Multi-step Methods for Stiff Systems. Acta Universitatis Apulensis. **30**, 171-188.
[29] Kleefeld B. and Martin-Vaquero J. (2016). SERK2v3: Solving Mildly Stiff Nonlinear Partial Differential Equations. Journal of Computational and Applied Mathematics. **299**, 194-206.
[30] Lambert J. D. (1972). Computational Methods in Ordinary Differential Equations. John Wiley Sons, New York.
[31] Schiesser W. E. and Griffiths G. W. (2009). A Compendium of Partial Differential Equation Models: Method of Lines Analysis with Matlab. Cambridge University Press, Cambridge.
[32] Smith G. D. (1977). Numerical Methods to Partial Differential Equations. Oxford University Press, Oxford.