



Contents lists available at ScienceDirect

Journal of King Saud University – Science

journal homepage: www.sciencedirect.com

Original article

Two-point block variable order step size multistep method for solving higher order ordinary differential equations directly



Ahmad Fadly Nurullah Rasedee^a, Mohammad Hasan Abdul Sathar^{b,*}, Siti Raihana Hamzah^c, Norizarina Ishak^c, Tze Jin Wong^d, Lee Feng Koo^d, Siti Nur Iqmal Ibrahim^e

^a Fakulti Ekonomi dan Muamalat, Universiti Sains Islam Malaysia, 71800 Nilai, Negari Sembilan, Malaysia

^b Centre of Foundation Studies for Agricultural Science, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

^c Fakulti Sains dan Teknologi, Universiti Sains Islam Malaysia, 71800 Nilai, Negari Sembilan Malaysia

^d Department of Basic Sciences and Engineering, Faculty of Agriculture and Food Science, Universiti Putra Malaysia, Bintulu Campus, 97008 Bintulu, Sarawak, Malaysia

^e Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

ARTICLE INFO

Article history:

Received 18 August 2020

Revised 13 December 2020

Accepted 7 February 2021

Available online 3 March 2021

2010 MSC:

34A12

34A34

65L06

65L05

Keywords:

Ordinary differential equations

Block

Multistep method

Variable order stepsize

ABSTRACT

The current research aims to provide a viable numerical method for solving difficult engineering and science problems which are in the form of higher order ordinary differential equations. The proposed method approximates these ordinary differential equations using Newton-Gregory backward difference polynomial in predictor–corrector mode. The predictor–corrector algorithm is then fitted with a variable order step size algorithm to reduce computational cost. The variable order stepsize algorithm allows the method to predetermine the preferred level of accuracy with the added advantage of less computational cost. The method is subsequently programmed with a two-point block formulation which can be altered for parallel programming. This research also discusses order and stepsize strategies of the variable order stepsize algorithm. Stability and convergence estimations of the method are also established. Numerical results obtained will validate the accuracy and efficiency of the method using various types of linear and nonlinear higher order ordinary differential equations.

© 2021 The Author(s). Published by Elsevier B.V. on behalf of King Saud University. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Problems in science and engineering are often modeled in the form of ordinary, partial and fractional differential equations depending on the type of problem. These problems may range from various field such as the interaction between chemicals as found in chemical kinetics to the mechanics of fluids which are often modelled in medicine and engineering. For recent analytical and numerical methods established for solving these differential applications, readers may refer to literature by Al-Jawary et al. (2020), Khataybeh et al. (2019), Mohd Ijam et al. (2020), Zhang et al.

(2020) and Rasedee et al. (2021) for ordinary differential equations, Colbrook et al. (2019), Fu et al. (2019) and Lehrenfeld and Olshanskii (2019) for partial differential equations and Goswami et al. (2019), Singh et al. (2019), Srivastava et al. (2020) and Veerasha et al. (2020) for fractional differential equations. The algorithm development provided in this research is to specifically cater a numerical method for solving higher order ordinary differential equations directly using variable order step size formulations. We begin the study with the evolution of the variable order step size algorithm based on the multistep method.

Previous numerical methods used to solve these higher order ordinary differential equations (ODEs) by reducing them to systems of first order ODEs were so efficient such that methods for solving ODEs directly were considered as robust. These methods were almost overlooked until (Krogh, 1968) revived the field of study with a modified version of the divided difference approach. He proposed that the back values of any point of the derivative were to be interpolated. In a d^{th} order ODE, the method used to interpolate the highest derivative value was referred as the Direct

* Corresponding author.

E-mail address: mohdhasan@upm.edu.my (M.H. Abdul Sathar).

Peer review under responsibility of King Saud University.



Production and hosting by Elsevier

Integration (DI). Krogh (1968) provide a comparison for two second order problems which was similar to Gear (1967) whom used a Nordsieck version of the multistep method. Suleiman (1979) extended works of Krogh by adapting the DI for estimating higher order derivatives (d^h order) which corresponds with nonstiff problems and a generalized backward differentiation method (GBDF) method for lower derivatives which corresponds with stiff problems. Omar and Suleiman (2005) adapted Suleiman's DI algorithm with a two-point explicit block algorithm. By doing so, Omar was able to reduce computational cost by providing a more efficient divided difference code. In efforts of solving higher order ODEs directly, Rasedee (2009) proposed a direct method for solving these higher order derivatives using a backward difference formulation. The backward difference method suggested by Rasedee was in effort to overcome the drawback of the DI method, as it requires calculating the integration coefficients at every step change which proves to be costly. The method proposed by Rasedee (2009) requires calculating the integration coefficients only once in the beginning of the algorithm and once more at the end, if deemed necessary.

This article is part of a research series influenced by the works of Suleiman (1979). He initiated a series of research based on Adams type difference formulation for solving ordinary differential equations (ODEs). In his research, Suleiman established the Direct Integration method (DI) for solving both, stiff and nonstiff initial value higher order ODEs directly. This method was formulated using a multistep method in divided difference form combined with variable order stepsize (VOS) capability. Using works by Suleiman as their foundation, authors such as Omar and Suleiman (2005), Majid and Suleiman (2006, 2007), Suleiman et al. (2011), Mohd Ijam et al. (2014) and Waeleh and Abdul Majid (2016) and Rasedee et al. (2016) have produced an extended variation for solving various types of differential problems.

The current research proposes a two-point block formulation in backward difference form (2PVOSBD) for solving both artificial and real life higher order ODE problems. Equipped with a VOS algorithm similar to Rasedee et al. (2014), the proposed method calculates the integration coefficients only once at the beginning and once at the end (if required) in contrast to the DI which requires calculating integration coefficients at every step change. The 2PVOSBD only requires calculating the integration coefficients a second time if $H \geq T_{end} - T_b$, where H as the current stepsize, T_{end} as the last point of the interval, and T_b as the current point.

When implementing a variable order stepsize strategy, the acceptance criteria is key. An unsuitable criteria may result in an increase in the total of steps or the loss of accuracy. The selection criteria is discussed in the upcoming sections. Following the appropriate selection criteria, strategies suggested by Lambert (1973) are used and integrated with the variable step size code proposed by Krogh (1973).

The current research also provides the discussion on the order, stability and convergence of the proposed method. For the variable order step size techniques and selection criterion implemented in this research, authors adapted techniques suggested by Gear (1967), Krogh (1968), Lambert (1973), Hall and Watt (1976) and Suleiman (1979). And in more recent works for solving higher order ODEs found in Zainuddin et al. (2016), Rasedee et al. (2017b, 2018a), Ibrahim et al. (2019) and Asnor et al. (2019) and others. The following are preliminary definitions that are used to determine order, stability and convergence of the 2PVOSBD.

2. Preliminaries

This section entails definitions used to determine the order, stability and convergence of the method.

Definition 2.1. The general linear multistep method is denoted by

$$\sum_{b=0}^k \alpha_b f_{n+b} = h \sum_{b=0}^k \beta_b \phi_{n+b}$$

Definition 2.2. The linear differential operator L associated with the linear multistep method is defined

$$L[f(t); h] := \sum_{i=0}^k [\alpha_i f(t + ih) - h \beta_i f'(t + ih)],$$

where $f(t)$ is an arbitrary function in $C^1[a, b]$.

Choose a function $f(t)$ to be differentiable as often as needed, and expand $f(t + ih)$ and $f'(t + ih)$ with respect to t , and arrange as

$$L[f(t); h] := C_0 f(t) + C_1 h f'(t) + \dots + C_q h^q z^{(q)}(t) + \dots$$

where C_i , $i = 0, 1, \dots, q, \dots$ are constants.

Definition 2.3. The linear multistep method and associated difference operator L defined are said to be of order p if in, $C_0 = C_1 = \dots = C_q = 0$, $C_{q+1} \neq 0$ where

$$\begin{aligned} C_0 &= \alpha_0 + \alpha_1 + \dots + \alpha_k \\ C_1 &= (\alpha_1 + 2\alpha_2 + \dots + k\alpha_k) - (\beta_1 + \beta_2 + \dots + \beta_k) \\ &\vdots \\ C_p &= \frac{1}{p} (\alpha_1 + 2\alpha_2^p + \dots + k^p \alpha_k) - \frac{1}{q-1} (\beta_1 + 2^{p-1} \beta_2 + \dots + k^{p-1} \beta_k) \end{aligned}$$

Definition 2.4. The block method is zero stable if the roots r_j , $j = 1, \dots, k$ of the first characteristic polynomial $\rho(r)$ is denoted by

$$\rho(r) = \det \left(\sum_{i=0}^m A_i r^{m-i} \right) = 0,$$

which satisfies the conditions $|r_j| \leq 1$ and the roots with $|r_j| = 1$, where the multiplicity does not exceed 2.

Definition 2.5. A Linear Multistep Method (LLM) is said to be consistent if the LLM is of order $p \geq 1$.

3. The higher order ordinary differential equation

Differential equations have been widely used to model various types of real-life applications, from natural phenomenon to man-made machinery. Examples of these higher order ODEs can found in classical mechanics of the two-body problem, chaotic motions in periodic-self excited oscillators to the elasticity of thin clamp plates.

The focus of this research is to establish a numerical method for solving higher order initial value ordinary differential equations. For the purpose of this research, consider the following higher order ODE

$$f^{(n)}(t) = \phi(t, f, f', f'', \dots, f^{(n-1)}). \quad (3.1)$$

The initial value condition is defined by $F(\alpha) = \tilde{\varepsilon}$ in the interval $\alpha \leq t \leq \beta$ where

$$F^T = \phi(f, f', f'', \dots, f^{(n-1)})$$

and

$$\varepsilon^T = (\varepsilon, \varepsilon', \varepsilon'', \dots, \varepsilon^{(n-1)}).$$

Conventionally, numerical methods would approximate (3.1) by reducing it to a system of first order ODEs. The equation is reduced by letting

$$f = w_1, \quad f' = w_2, \quad f'' = w_3, \quad \dots, \quad f^{(n-1)} = w_d$$

with the initial conditions

$$w^T = (w_1, w_2, \dots, w_d)$$

and

$$F^T(t, w) = (w_1, w_2, \dots, w_d, \phi)$$

which reduces (3.1) to

$$w' = F(t, w)$$

subjected to the initial conditions $w(a) = \varepsilon$.

Direct methods versus reduction to first order systems have been extensively discussed by Rutishauser (1960), Collatz (1960), Henrici (1962) and Gear (1967). These opinions may vary based on individual preferences. In the case of Henrici (1962), the author favours reduction methods due to the large global error of direct methods even though lower order derivatives have small local truncation error, whereas Collatz (1960) and Gear (1967) prefer direct methods. Rutishauser (1960) provided proof that the n th linear order ODE was solved using a direct method (Euler extended method) had a global error of order 1, which was equal to the global error of the reduction method (Euler method). A N order direct method algorithm requires only storing and updating one set of back values, where a N reduction method needs N set of back values. Due to the fact that the accuracy of each derivative is of the same order thus, the direct method requires controlling the local error of only $(N - 1)^{th}$ derivative because it offers the lowest accuracy $O(h)$ opposed to controlling the local error of each derivative as required by the reduction method.

The method proposed in this research approximates (3.1) directly with minimal loss of accuracy, if not any. The succeeding section will provide detailed derivation of the proposed method.

4. The two-point block method

This section explains derivation of both first and second point of the two-point block method. To formulate a two-point predictor-corrector (PeCe) block method with a variable order step size backward difference algorithm, elements such as explicit-implicit integration coefficients and order stepsize strategy are necessary.

Firstly, note that the 2PVOSBD method is formulated in predictor-corrector mode similar to the Adam-Basthforth-Moulton formulation. The derivation begins with integrating the higher order ODE as defined in (3.1). By integrating (3.1) once, $f^{(n)}$ becomes

$$f^{(n-1)}(t_{i+1}) = f^{(n-1)}(t_i) + \int_{t_i}^{t_{i+1}} \sum_{m=0}^k (-1)^i \binom{-s}{i} \nabla^i \phi_{n+(1-p)} dt, \quad p = 0, 1 \quad (4.2)$$

$$f^{(n-1)}(t_{i+2}) = f^{(n-1)}(t_i) + \int_{t_i}^{t_{i+2}} \sum_{m=0}^k (-1)^i \binom{-s}{i} \nabla^i \phi_{n+(2-p)} dt, \quad p = 0, 2. \quad (4.3)$$

From Eqs. (4.2) and (4.3), $f^{(n-1)}(t_{i+1})$ represents the first block and $f^{(n-1)}(t_{i+2})$ as the second block of the 2PVOSBD method. When $p = 0$, both Eqs. (4.2) and (4.3) indicates the corrector equations whereas when p is otherwise, the equations takes form of the pre-

dictor. The following steps are taken to establish the explicit coefficients for the predictor. First, the limit of integration is changed and dt is replaced by hds and substituted into the two-point predictor formula

$$f^{(n-1)}(t_{i+b}) = f^{(n-1)}(t_i) + \int_0^b (b-s) \sum_{m=0}^k (-1)^m \binom{-s}{m} \nabla^m \phi_n hds, \quad b = 1, 2. \quad (4.4)$$

Then, denoting $\gamma_{bj,i}^p$ by

$$\gamma_{bj,i}^p = (-1)^i \int_0^b \binom{-s}{i} ds,$$

the predictor can be reformulated as follows:

$$f^{(n-1)}(t_{i+b}) = f^{(n-1)}(t_i) + h \sum_{m=0}^k \gamma_{bj,i}^p \nabla^m \phi_n ds. \quad (4.5)$$

Next, formulating the corrector formula begins in a similar manner as the predictor with the exception of changing the limit of integration from $-b$ to 0 thus,

$$f^{(n-1)}(t_{i+b}) = f^{(n-1)}(t_i) + \int_{-b}^0 (-s) \sum_{m=0}^k (-1)^m \binom{-s}{m} \nabla^m \phi_{n+b} hds.$$

Now, let $\gamma_{bj,i}^c$ as

$$\gamma_{bj,i}^c = (-1)^i \int_{-b}^0 \binom{-s}{i} ds,$$

hence (4.5) can be rewritten as

$$f^{(n-1)}(t_{i+b}) = f^{(n-1)}(t_i) + h \sum_{m=0}^k \gamma_{bj,i}^c \nabla^m \phi_n ds. \quad (4.6)$$

As for a N^{th} order ODE, the predictor-corrector can be deduced by integrating (3.1) by N -fold integrals and applying similar requirements used in the first integral, which yields the general formulation;

Predictor:

$$f^{(d-j)}(x_{n+b}) = \sum_{i=0}^{j-1} \frac{h^i}{i!} f^{(d-j+i)}(x_n) + h^j \sum_{i=0}^{k-1} \gamma_{bj,i}^p \nabla^i \phi_n, \quad (4.7)$$

Corrector:

$$f^{(d-j)}(x_{n+b}) = \sum_{i=0}^{j-1} \frac{h^i}{i!} f^{(d-j+i)}(x_n) + h^j \sum_{i=0}^{k-1} \gamma_{bj,i}^c \nabla^i \phi_{n+b}. \quad (4.8)$$

with the respective explicit, $\gamma_{bj,i}^p$ and implicit, $\gamma_{bj,i}^c$ integration coefficients as follows

Explicit:

$$\gamma_{bj,i}^p = (-1)^i \int_0^b \frac{(b-s)^{d-1}}{(d-1)!} \binom{-s}{i} ds,$$

Implicit:

$$\gamma_{bj,i}^c = (-1)^i \int_{-b}^0 \frac{(-s)^{d-1}}{(d-1)!} \binom{-s}{i} ds.$$

5. Integration coefficients

The current section provides derivation for explicit and implicit coefficients. Firstly, let the first order explicit, $G_{b,1}^p(t)$ and implicit $G_{b,1}^c(t)$ generating functions as

$$G_{b,1}^p(t) = \sum_{i=0}^{\infty} \gamma_{1,d,i}^p t^i$$

and

$$G_{b,1}^c(t) = \sum_{i=0}^{\infty} \gamma_{1,d,i}^c t^i.$$

Next, by substituting $\gamma_{b,d,i}^p$ into $G_{(d)}^p(t)$, the explicit generating function can be rewritten as

$$G_{b,1}^p(t) = \sum_{i=0}^{\infty} (-t)^i \int_0^b (b-s) \binom{-s}{i} ds$$

which can be simplified as

$$G_{b,1}^p(t) = \int_0^b (1-t)^{(-s)} ds. \quad (5.9)$$

By solving the integral in (5.9) and mathematical induction, $G_{b,1}^p(t)$ can be represented as

$$G_{b,1}^p(t) = - \left[\frac{(1-t)^{-b}}{\log(1-t)} - \frac{1}{\log(1-t)} \right]. \quad (5.10)$$

Then by substituting

$$\gamma_{b,j,i}^c = (-1)^i \int_{-b}^0 \binom{-s}{i} ds,$$

into $G_{b,1}^c(t)$, reveals the implicit generating function in form of

$$G_{b,1}^c(t) = - \left[\frac{1}{\log(1-t)} - \frac{(1-t)^b}{\log(1-t)} \right]. \quad (5.11)$$

Through mathematical induction, these explicit and implicit generating functions can be generalized and constructed as

$$G_{(d)}^p(t) = \frac{1}{(d-1)!} \left[\frac{b^{d-1}}{\log(1-t)} - \frac{(d-1)! G_{(d-1)}^p(t)}{\log(1-t)} \right].$$

$$G_{(d)}^c(t) = \frac{1}{(d-1)!} \left[\frac{b^{d-1} (1-t)^b}{\log(1-t)} - \frac{(d-1)! G_{(d-1)}^c(t)}{\log(1-t)} \right].$$

Integration coefficients are subsequently derived from the generating functions. By expanding then rearranging these generating functions, the relationship between coefficients of different orders yields the following

Explicit Coefficients:

$$\gamma_{b,d,0}^p = \gamma_{b,d-1,1}^p, \quad \gamma_{b,d,k}^p = \gamma_{b,d-1,k+1}^p - \sum_{i=0}^{k-1} \left(\frac{\gamma_{b,d,i}^p}{k-i+1} \right), \quad k = 1, 2, \dots$$

Implicit Coefficients:

Table 1
Explicit integration coefficients.

k	0	1	2	3	4	5	6	7
$\gamma_{1,k}^p$	1	1/2	5/12	3/8	251/720	95/288	19087/60480	5257/17280
$\gamma_{2,k}^p$	2	2	7/3	8/3	269/90	33/10	13613/3780	736/189

Table 2
Implicit integration coefficients.

k	0	1	2	3	4	5	6	7
$\gamma_{1,k}^c$	1	-1/2	-1/12	-1/24	-19/720	-3/160	-863/60480	-275/24192
$\gamma_{2,k}^c$	2	-2	1/3	0	-1/90	-1/90	-37/3780	-8/945

$$\gamma_{b,d,0}^c = \gamma_{b,d-1,1}^c, \quad \gamma_{b,d,k}^c = \gamma_{b,d-1,k+1}^c - \sum_{i=0}^{k-1} \left(\frac{\gamma_{b,d,i}^c}{k-i+1} \right), \quad k = 1, 2, \dots$$

Inline with the aim of this research to reduce computational cost, a recursive relationship between explicit and implicit coefficients is established. The relationship obtained begins in terms of the generating functions. Referring to Eqs. (5.10) and (5.11), which are the first order generating functions

$$G_{b,1}^p(t) = - \left[\frac{(1-t)^{-b}}{\log(1-t)} - \frac{1}{\log(1-t)} \right]. \quad (5.12)$$

$$G_{b,1}^c(t) = - \left[\frac{1}{\log(1-t)} - \frac{(1-t)^b}{\log(1-t)} \right]. \quad (5.13)$$

The explicit generating function, $G_{b,1}^p(t)$ in ((5.12) be rearranged as

$$G_{b,1}^p(t) = -(1-t)^{-b} \left[\frac{1}{\log(1-t)} - \frac{(1-t)^b}{\log(1-t)} \right], \quad (5.14)$$

then substituting (5.13) into (5.14) produces the following recursive relationship between explicit and implicit generating functions

$$(1-t)^b G_{b,1}^p(t) = G_{b,1}^c(t)$$

subsequently establish the relationship between coefficients

$$(1-t)^b \sum_{i=0}^{\infty} \gamma_{b,j,i}^p t^i = \sum_{i=0}^{\infty} \gamma_{b,j,i}^c t^i. \quad (5.15)$$

A few examples of the integration coefficients extracted from (5.15) are presented in Tables 1 and 2 below.

6. Order and stability

Various researchers have studied conditions of zero stability and the order of a block method which can be found in works of Watts and Shampine (1972), Mohd Ijam et al. (2018), Mohd Ijam and Ibrahim (2019) and Rasedee et al. (2021a) and etc. Discussions on order and stability of the proposed method applies definitions which are similar to Ola Fatunla (1991).

6.1. Order

The current section will show how the order of the method is established for the predictor in (4.7) and corrector in (4.8) when the given $k = 7$. For the purpose of this research, let consider the case where $d = 1$. The derivation begins with establishing the order method for the predictor. By extracting the explicit coefficients, the predicted first and second point can be expanded and written as

$$f_{n+1} = f_n + h \left(\frac{198721}{60480} \phi_n - \frac{9301}{1260} \phi_{n-1} + \frac{235183}{20160} \phi_{n-2} - \frac{10754}{945} \phi_{n-3} + \frac{135713}{20160} \phi_{n-4} - \frac{5063}{2520} \phi_{n-5} + \frac{19087}{60480} \phi_{n-6} \right) \quad (6.16)$$

$$f_{n+2} = f_n + h \left(\frac{14281}{756} \phi_n - \frac{4649}{126} \phi_{n-1} + \frac{145261}{1260} \phi_{n-2} - \frac{113068}{945} \phi_{n-3} + \frac{92621}{1260} \phi_{n-4} - \frac{7846}{315} \phi_{n-5} + \frac{13613}{3780} \phi_{n-6} \right) \quad (6.17)$$

To start off the derivation of the order method, (6.16) and (6.17) are defined in the form of matrices which satisfy Definition 2.1. Let

$$\sum_{i=0}^4 A_i F_{m+i-4}^p = h \sum_{i=0}^4 B_i \Phi_{m+i-4}^p$$

be the general matrix where

$$\sum_{i=0}^4 A_i F_{m+i-4}^p = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} f_{n-7} \\ f_{n-6} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} f_{n-5} \\ f_{n-4} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} f_{n-3} \\ f_{n-2} \end{pmatrix} + \begin{pmatrix} 0 & -1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} f_{n-1} \\ f_n \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} f_{n+1} \\ f_{n+2} \end{pmatrix},$$

and

$$\sum_{i=0}^4 B_i \Phi_{m+i-4}^p = \begin{pmatrix} 0 & \frac{19087}{60480} \\ 0 & \frac{13613}{3780} \end{pmatrix} \begin{pmatrix} \phi_{n-7} \\ \phi_{n-6} \end{pmatrix} + \begin{pmatrix} -\frac{5603}{2520} & \frac{135713}{20160} \\ -\frac{7846}{315} & \frac{92621}{1260} \end{pmatrix} \begin{pmatrix} \phi_{n-5} \\ \phi_{n-4} \end{pmatrix} + \begin{pmatrix} -\frac{10754}{945} & \frac{235183}{20160} \\ -\frac{113068}{945} & \frac{145261}{1260} \end{pmatrix} \begin{pmatrix} \phi_{n-3} \\ \phi_{n-2} \end{pmatrix} + \begin{pmatrix} -\frac{18637}{2520} & \frac{198721}{60480} \\ -\frac{4078}{63} & \frac{14281}{756} \end{pmatrix} \begin{pmatrix} \phi_{n-1} \\ \phi_n \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \phi_{n+1} \\ \phi_{n+2} \end{pmatrix}$$

Next, the coefficients are separated into the following block matrices

$$a_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$a_5 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_6 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad a_7 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \quad a_8 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad a_9 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$b_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad b_1 = \begin{pmatrix} \frac{19087}{60480} \\ \frac{13613}{3780} \end{pmatrix}, \quad b_2 = \begin{pmatrix} -\frac{5603}{2520} \\ \frac{135713}{20160} \end{pmatrix}, \quad b_3 = \begin{pmatrix} \frac{135713}{20160} \\ \frac{92621}{1260} \end{pmatrix},$$

$$b_4 = \begin{pmatrix} -\frac{10754}{945} \\ -\frac{113068}{945} \end{pmatrix}, \quad b_5 = \begin{pmatrix} \frac{235183}{20160} \\ \frac{145261}{1260} \end{pmatrix}, \quad b_6 = \begin{pmatrix} -\frac{18637}{2520} \\ -\frac{4078}{63} \end{pmatrix},$$

$$b_7 = \begin{pmatrix} \frac{198721}{60480} \\ \frac{14281}{756} \end{pmatrix}, \quad b_8 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad b_9 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Then applying the coefficients in Definition 2.3, the result is

$$C_0 = C_1 = C_2 = \dots = C_7 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad C_8 = \begin{pmatrix} \frac{515281}{1693440} \\ \frac{736}{189} \end{pmatrix}.$$

From the result obtained, it can be concluded that the predictor formula when $k = 7$, is of order 7 with an error constant of

$$C_8 = \begin{pmatrix} \frac{515281}{1693440} \\ \frac{736}{189} \end{pmatrix}$$

For the following corrector,

$$f_{n+1} = f_n + h \left(\frac{5257}{17280} \phi_{n+1} + \frac{139849}{120960} \phi_n - \frac{4511}{4480} \phi_{n-1} + \frac{123133}{120960} \phi_{n-2} - \frac{88547}{120960} \phi_{n-3} + \frac{1537}{4480} \phi_{n-4} - \frac{11351}{120960} \phi_{n-5} + \frac{275}{24192} \phi_{n-6} \right) \quad (6.18)$$

$$f_{n+2} = f_n + h \left(\frac{41}{140} \phi_{n+2} + \frac{1466}{945} \phi_{n+1} - \frac{71}{420} \phi_n + \frac{68}{105} \phi_{n-1} - \frac{1927}{3780} \phi_{n-2} + \frac{26}{105} \phi_{n-3} - \frac{29}{420} \phi_{n-4} + \frac{8}{945} \phi_{n-5} \right) \quad (6.19)$$

the result can be derived similarly to the predictor formula. The result show that when $k = 7$, the corrector formulas is of order 8 with an error constant of

$$C_9 = \begin{pmatrix} -\frac{33952}{3628800} \\ -\frac{119}{16200} \end{pmatrix}$$

6.2. Stability

The viability of a linear multistep method is subject to certain conditions governed by zero stability. Similarly to works discussed in Hall and Watt (1976), the stability of the two-point predictor can be obtained by applying Eqs. (6.16) and (6.17) to the standard linear test problem,

$$f' = \lambda f.$$

Firstly, the predictor formula is amended in the form of block matrices which is substituted into the standard linear test problem which then expressed as

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} f_{n+1} \\ f_{n+2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} f_{n-1} \\ f_n \end{pmatrix} h\lambda + \begin{pmatrix} -\frac{18637}{2520} & \frac{198721}{60480} \\ -\frac{4078}{63} & \frac{14281}{756} \end{pmatrix} \begin{pmatrix} f_{n-1} \\ f_n \end{pmatrix} h\lambda + \begin{pmatrix} -\frac{10754}{945} & \frac{235183}{20160} \\ -\frac{113068}{945} & \frac{145261}{1260} \end{pmatrix} \begin{pmatrix} f_{n-3} \\ f_{n-2} \end{pmatrix} h\lambda + \begin{pmatrix} -\frac{5603}{2520} & \frac{135713}{20160} \\ -\frac{7846}{315} & \frac{92621}{1260} \end{pmatrix} \begin{pmatrix} f_{n-5} \\ f_{n-4} \end{pmatrix} h\lambda + \begin{pmatrix} 0 & \frac{19087}{60480} \\ 0 & \frac{13613}{3780} \end{pmatrix} \begin{pmatrix} f_{n-7} \\ f_{n-6} \end{pmatrix} h\lambda \quad (6.20)$$

therefore, it can be represented as equivalent to

$$AF_i - (B + C)F_{i-1} - DY_{i-2} - EF_{i-3} - GF_{i-4} = 0, \quad n = 2i.$$

For the purpose of showing stability, (6.20) is reformulated in form of the stability polynomial which is dictated by, $R(t; \Lambda)$,

$$R(t; \Lambda) = \det \left(\sum_{i=0}^m A_i t^{m-i} \right), \quad \Lambda = h\lambda \quad (6.21)$$

where

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_1 = -\begin{pmatrix} -\frac{18637}{2520} & 1 + \frac{198721}{60480} \\ -\frac{4078}{63} & 1 + \frac{14281}{756} \end{pmatrix},$$

$$A_2 = -\begin{pmatrix} -\frac{10754}{945} & \frac{235183}{20160} \\ -\frac{113068}{945} & \frac{145261}{1260} \end{pmatrix}, \quad A_3 = -\begin{pmatrix} -\frac{5603}{2520} & \frac{135713}{20160} \\ -\frac{7846}{315} & \frac{92621}{1260} \end{pmatrix},$$

$$A_4 = -\begin{pmatrix} 0 & \frac{19087}{60480} \\ 0 & \frac{13613}{3780} \end{pmatrix} \quad (6.22)$$

Substituting (6.22) into (6.21), attains the following stability polynomial

$$R(t; \Lambda) = t^8 - \left(1 + \frac{86899}{7560} \Lambda \right) t^7 + \left(\frac{9931223}{136080} \Lambda^2 - \frac{70417}{1512} \Lambda \right) t^6 + \left(\frac{54893593}{680400} \Lambda^2 + \frac{55919}{1512} \Lambda \right) t^5 + \left(\frac{5374679}{340200} \Lambda^2 + \frac{144269}{7560} \Lambda \right) t^4 - \frac{1032431}{680400} t^3 \Lambda^2 + \frac{688823}{680400} t^2 \Lambda^2 - \frac{99667}{680400} t \Lambda^2$$

In order to determine zero stability, let $\Lambda = 0$, which yields

$$R(t; \Lambda) = t^8 - t^7$$

with the roots $t_0 = t_1 = \dots = t_6 = 0$, $t_7 = 1$. Thus by using [Definition 2.4](#), the predictor formula is zero stable.

Whereas for the corrector, the stability polynomial is established as follows

$$\begin{aligned} R(t; \Lambda) = & \left(1 - \frac{72223}{120960}\Lambda + \frac{30791}{345600}\Lambda^2\right)t^8 \\ & - \left(1 + \frac{239}{3360}t^7\Lambda + \frac{122302297}{57153600}\Lambda^2\right)t^7 \\ & - \left(\frac{24961}{60480}\Lambda + \frac{128389687}{50803200}\Lambda^2\right)t^6 \\ & - \left(\frac{2642501}{3212160}\Lambda + \frac{10775593693}{12141964800}\Lambda^2\right)t^5 \\ & - \left(\frac{5807}{60228}\Lambda + \frac{2409485711}{54638841600}\Lambda^2\right)t^4 \\ & - \frac{202192573}{36425894400}t^3\Lambda^2 + \frac{536771}{1517745600}t^2\Lambda^2 \\ & - \frac{55}{571536}t\Lambda^2 \end{aligned} \quad (6.23)$$

When $\Lambda = 0$, the stability polynomial yields

$$R(t; \Lambda) = t^8 - t^7 \quad (6.24)$$

with the roots $t_0 = t_1 = \dots = t_6 = 0$, $t_7 = 1$. Hence it proves that the corrector formula as zero stable.

7. Convergence of the backward difference method

As previously shown by [Suleiman \(1993\)](#), there are certain conditions that need to be met to show the convergence of the backward difference method.

Theorem 7.1. *The necessary conditions for the linear multistep method (4.7) and (4.8) to be convergent are*

- i) the method must be zero.
- ii) the method must be consistent.

Proof of the theorem can be found in [Suleiman \(1993\)](#).

In the previous section, both the predictor and corrector satisfies conditions in [Definition 2.4](#), making the method zero stable; whereas the order method as previously shown in [Definition 2.3](#), with the predictor

$$C_0 = C_1 = C_2 = \dots = C_7 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad C_8 \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

and the corrector

$$C_0 = C_1 = C_2 = \dots = C_8 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad C_9 \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

are consistent with [Definition 2.5](#) of order of 8 and 9 respectively. Satisfying both conditions i) and ii) of [Theorem 7.1](#) shows that the method to be convergent.

8. Variable order step size strategies

The literature on the variable order stepsize codes show an incomplete theory with numerous strategies of order and stepsize selection are given. Firstly, varying the order. The variable strategies are dependent on the amount of back values stored. With the appropriated amount of back values, the order can simply be increased by retaining the values of previous steps and can be decreased by disregarding unnecessary values. Previous researches have shown that

the order strategies which are unbiased to lower order is efficient in an Adams based code for solving nonstiff problems.

As mentioned earlier, the acceptance criteria in a variable order strategy is crucial. In this article, the second step evaluation is determined on whether the estimated error,

$$E_k^{(d-p)} = h_{b,d-p,k}^d \nabla^k \phi_{n+1}$$

satisfies the local accuracy requirement,

$$\Theta_{n+1} |E_k^{(d-p)}| < TOL, \quad \Theta_{n+1} = \frac{1}{(A + B + P_n)}$$

where A and B determine the type of error test, which is discussed later in the numerical results section. For reason of simplicity, the error estimate $E_k^{(d-p)}$ is denoted as E_k , where $k = n - 2, n - 1, n, n + 1$. The order condition is set for $n > 2$, $\max\{|E_{n-1}|, |E_{n-2}|\} \leq |E_n|$ and if both conditions $|E_{n-1}| \leq 0.5|E_n|$ and $n = 2$ are satisfied, the order is reduced by 1. If E_{n+1} is available, the condition to reduce the order is when $n > 1$ and $|E_{n-1}| \leq \min\{|E_n|, |E_{n+1}|\}$ is satisfied. For the latter, the algorithm must first complete $n + 1$ successful steps in constant stepsize, only then for $n > 1$ where $|E_{n+1}| < |E_n| < \max\{|E_{n-1}|, |E_{n-2}|\}$, and both $|E_{n+1}| < 0.5|E_n|$ and $n = 1$ are satisfied will the order be increased.

As for varying the stepsize, this research adopts the halving and doubling of the stepsize algorithm suggested by [Krogh \(1973\)](#). Although doubling the stepsize might seem necessary because the previous back values of $\phi_{n-2}, \phi_{n-4}, \dots, \phi_{n-2k+2}$ can be used, Krogh argues that such technique will contribute to the loss of accuracy. In fact, doubling the stepsize after every evaluation may also have a toll on the accuracy. This is when halving the stepsize is required. In halving the step size algorithm, new back values are obtained. The use of back values in the form of $\phi_{n-\frac{1}{2}}, \phi_{n-1}, \dots, \phi_{n-\frac{k+1}{2}}$ has shown to be more accurate. For more details on the algorithm, reader may refer to [Rasedee \(2009\)](#) and [Rasedee et al. \(2014\)](#). [Fig. 1](#) illustrates the general flow of the proposed 2PVOSBD algorithm.

9. Numerical results and discussions

As numerical method becomes more robust, the accuracy of a method alone is no longer sufficient hence, the need for efficiency. When dealing with complex problems that require massive calculation, computational cost becomes a concern. The current research highlights the efficiency of proposed method. With little to none loss of accuracy, the 2PVOSBD offers an accuracy to total step ratio which proves to be efficient, similar as proposed in [Rasedee \(2009\)](#).

The 2PVOSBD method is tested with a combination of linear and nonlinear higher order initial value ODEs. The selected test problems are also inclusive of artificial and real-life, practical ODEs. The approximation obtained by the 2PVOSBD was compared with approximations of 1PVOSBD and DI methods which were reconstructed for the purpose of this research. Due to its divided difference based algorithm, the DI method was considered as a suitable counterpart of the 2PVOSBD backward difference based algorithm. The comparison is based on accuracy, total steps and efficiency of the methods. To compare approximated solution, the error solution in this research is defined by

$$Err_n = \left| \frac{(f_i)_n - f(t_i)_n}{A + B(f(t_i)_n)} \right|$$

where $(f_i)_n$ is the n^{th} component of the exact solution and $f(t_i)_n$ is the n^{th} component of the approximated solution of f . There are 3

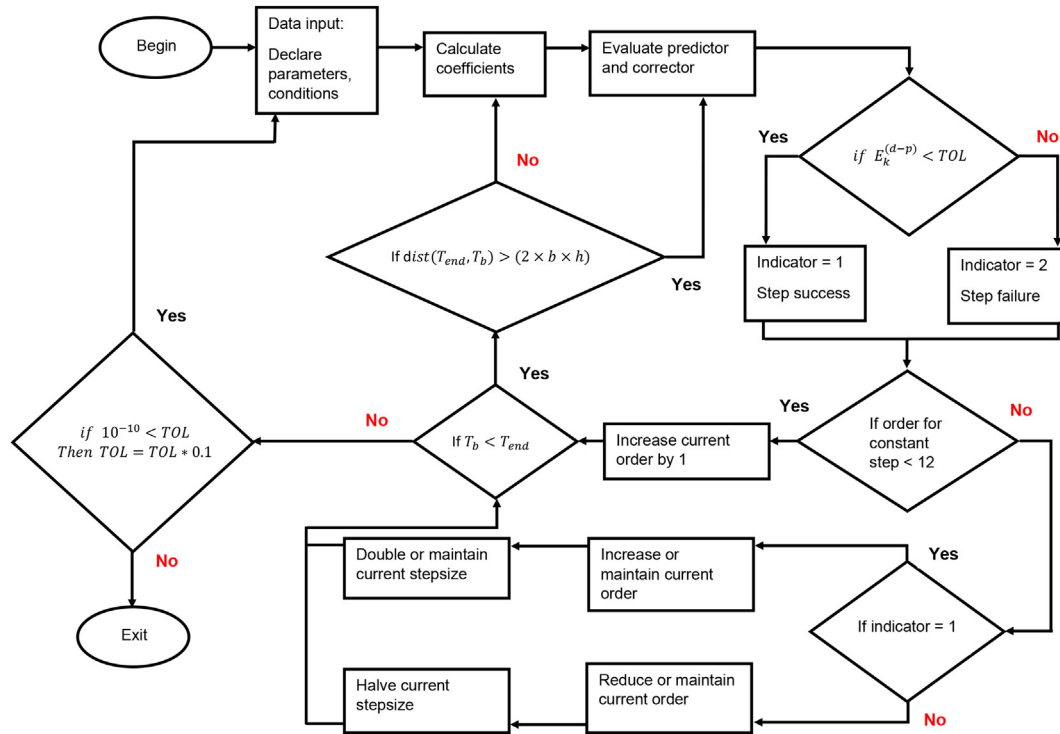


Fig. 1. The general flowchart for the 2PVOSBD algorithm.

types of error test provided in the algorithm, where the most suitable estimation is selected. The choice of error tests are, absolute error when the corresponding $A = 1, B = 0$, relative error when $A = 0, B = 1$. and mixed error when $A = 1, B = 1$. For the current research, the mixed error test was deemed best suited. On the other hand, the maximum error (MaxErr) is defined as

$$\text{MaxErr} = \max_{1 \leq j \leq \text{SuccStp}} \left(\max_{1 \leq n \leq N} (\text{Err}_n) \right)$$

and the average error (AvErr)

$$\text{AvErr} = \frac{\sum_{i=1}^{\text{SuccStp}} \sum_{n=1}^N \sum_{l=1}^B \text{Err}_n}{(B * N * \text{SuccStp})}$$

with N as the number of equations and B as the number of blocks. Contrary, the efficiency of the methods is defined as the ratio of accuracy compared to the corresponding total steps. These are the abbreviations that are used in the current section:

PeCe	predictor–corrector
ODE	ordinary differential equation
TOL	tolerance level
H	stepsize
TStep	total steps
MaxErr	overall maximum error
AvErr	average error
MTD	method used
DI	Direct Integration method
1PVOSBD	One-Point Block VOS backward difference method
2PVOSBD	Two-Point Block VOS backward difference method

The following are test problems that were selected for comparisons.

Problem 1. (Non linear 5th order ODE): The equation $f^{(v)} - 2f'f'' + ff^{(iv)} + f'f''' = (t^2 - 2t - 3)e^t - 8t$ for $0 \leq t \leq 2$ with initial value conditions $f(0) = 1, f'(0) = 1, f''(0) = 3, f'''(0) = 1, f^{(iv)}(0) = 1$ and $f(t) = e^t + t^2$ as the exact solution. Source: [Waeleh and Abdul Majid \(2016\)](#).

Problem 2. (Non linear 5th order ODE): The equation $f^{(v)} - 6(2(f')^3 + 6ff'f'' + f^2f''') = 0$ for $1 \leq t \leq 3$ with initial value conditions $f(1) = 1, f'(1) = -1, f''(1) = 2, f'''(1) = -6, f^{(iv)}(1) = 24$ and $f(t) = \frac{1}{t}$ as the exact solution. Source: [Waeleh and Abdul Majid \(2016\)](#).

Problem 3. (Artificial problem): The equation $f^{(8)} - f = 0$ for $0 \leq t \leq 100$ with initial value conditions $f(0) = f'(0) = \dots = f^{(7)}(0) = 0$ and $f(t) = e^t$ as the exact solution. Source: [Suleiman \(1979\)](#).

Problem 4. (4th order ODE) The equation $f^{(iv)} + 0.1f^{(v)} + 5f^{(iv)} + 0.5f''' + 4f'' - 0.4f' = e^{-t}$ for $0 \leq t \leq 16\pi$ with initial value conditions $f(0) = 3\frac{1}{9}, f'(0) = 2.9 - A, f''(0) = -4.99 + A, f'''(0) = -9.001 - A, f^{(iv)}(0) = 17.0001 + A, f^{(v)}(0) = 32.99999 - A$ where $A = \frac{1}{9}$ and $f(t) = \cos t + \sin t + \cos 2t + \sin 2t + e^{-0.1t} + \frac{1}{9}e^{-t}$ as the exact solution.

Problem 5. (Van Der Pol equation): The equation $f'' - \mu(1 - f^2)f' + f = 0$ for $0 \leq t \leq 15\pi$ with initial value conditions $f(0) = \alpha_1, f'(0) = \alpha_2$.

Table 3

Numerical result of the 2PVOSBD in constant order and stepsize mode for Problems 1 and 2.

H	Problem 1		Problem 2	
	TSteps	MaxErr	TSteps	MaxErr
10^{-1}	10	1.33721(−5)	10	3.05861(−1)
10^{-2}	100	2.10269(−8)	100	8.9045(−4)
10^{-3}	1000	2.20153(−11)	1000	9.64991(−7)
10^{-4}	10000	6.72519(−14)	10000	9.727174(−10)
10^{-5}	100000	3.39272(−12)	100000	1.101204(−12)

Table 4

Numerical result of the 2PVOSBD in constant order and stepsize mode for Problems 3 and 4.

H	Problem 3		Problem 4	
	TSteps	MaxErr	TSteps	MaxErr
10^{-1}	500	5.62642(−5)	252	2.75520(−1)
10^{-2}	5000	5.13835(−8)	2514	2.75520(−4)
10^{-3}	50000	7.40008(−11)	25133	2.68345(−7)
10^{-4}	500000	1.07478(−9)	251328	2513275(−10)
10^{-5}	5000000	9.34339(−9)	2513275	6.31914(−9)

Table 5

Comparisons of total steps and accuracy for Problems 1 and 2.

TOL	MTD		Problem 1		Problem 2	
10^{-2}	DI	10	6.17871(−6)	6.21332(−7)	16	3.19948(−1)
	VOSBD	14	1.17684(−5)	8.61228(−7)	23	1.81084(−1)
	2PBVOS	15	3.76917(−4)	2.82770(−5)	23	1.57933(−2)
10^{-4}	DI	13	3.67561(−6)	3.53799(−7)	24	1.99444(−4)
	VOSBD	18	1.52822(−5)	9.91370(−7)	31	8.90574(−3)
	2PBVOS	27	5.46305(−4)	1.94502(−4)	25	1.72907(−5)
10^{-6}	DI	17	2.37716(−7)	2.28106(−8)	33	2.93390(−4)
	VOSBD	21	3.57616(−7)	3.32091(−8)	43	8.34141(−4)
	2PBVOS	26	5.70940(−6)	1.30323(−6)	29	6.92038(−6)
10^{-8}	DI	23	2.56338(−8)	3.16854(−9)	46	1.50674(−5)
	VOSBD	32	4.60778(−8)	1.02584(−8)	57	7.57985(−8)
	2PBVOS	32	1.20022(−7)	3.82540(−8)	39	1.297591(−7)
10^{-10}	DI	31	2.29642(−9)	3.07931(−10)	84	3.91504(−8)
	VOSBD	55	2.20703(−8)	6.82589(−9)	71	1.35719(−7)
	2PBVOS	49	2.58567(−9)	2.07796(−9)	58	4.02752(−7)
10^{-10}	DI	31	2.29642(−9)	3.07931(−10)	84	3.91504(−8)
	VOSBD	55	2.20703(−8)	6.82589(−9)	71	1.35719(−7)
	2PBVOS	49	2.58567(−9)	2.07796(−9)	58	4.02752(−7)

Table 6

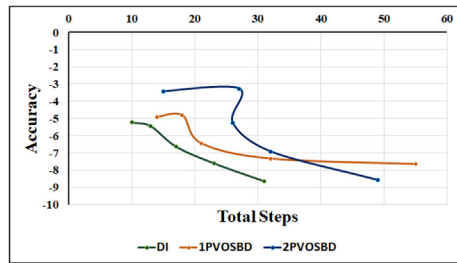
Comparisons of total steps and accuracy for Problems 3 and 4.

TOL	MTD	TSteps	Problem 3 MaxErr	AvErr	TSteps	Problem 4 MaxErr	AvErr
10^{-2}	DI	96	4.99681(−3)	2.47698(−3)	128	2.19551(0)	6.89295(−1)
	VOSBD	173	1.01436(−4)	8.58874(−5)	126	1.37312(0)	6.78583(−1)
	2PBVOS	92	1.74811(−4)	1.46155(−4)	103	2.43499(0)	1.46793(0)
10^{-4}	DI	141	1.79264(−4)	1.19952(−4)	204	1.00719(0)	7.66888(−1)
	VOSBD	142	2.60404(−4)	1.89056(−4)	202	2.73723(−2)	5.73163(−3)
	2PBVOS	138	1.14847(−5)	7.75051(−6)	156	7.39295(−1)	3.47230(−1)
10^{-6}	DI	214	1.18826(−5)	9.01852(−6)	51	1.30145(0)	NAN
	VOSBD	218	1.56415(−6)	1.09680(−6)	296	8.32079(−3)	2.91860(−3)
	2PBVOS	217	4.19170(−7)	4.44841(−7)	248	6.26771(−3)	7.77267(−4)
10^{-8}	DI	335	3.11508(−8)	2.58590(−8)	409	1.17334(0)	5.67692(−1)
	VOSBD	336	3.64286(−8)	1.21667(−8)	434	1.12434(−4)	2.59048(−5)
	2PBVOS	335	1.44458(−8)	1.59774(−8)	348	7.573041(−5)	9.25471(−6)
10^{-10}	DI	516	8.37294(−10)	6.51148(−10)	384	1.35703(−7)	4.00835(−8)
	VOSBD	517	3.30834(−9)	2.52921(−9)	393	8.39002(−8)	1.93736(−8)
	2PBVOS	521	3.10169(−10)	1.32179(−10)	385	5.10680(−6)	2.09405(−7)

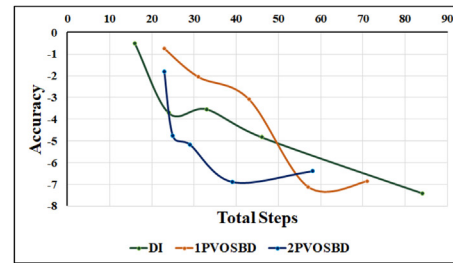
Tables 3 and 4 are the approximated results for Problems 1–4 using the 2PVOSBD, but with the absence of the variable order stepsize component. The result presented compare the constant stepsizes, $H = 10^{-1}$, $H = 10^{-2}$, $H = 10^{-3}$, $H = 10^{-4}$ and $H = 10^{-5}$. Starting with Euler's method, the algorithm increase the back values (order) each step until the total of 12 back values were obtained. Then, the algorithm completed its estimation using 12 back values for the remaining steps.

Table 5 presents the numerical approximation of the DI, 1PVOSBD and 2PVOSBD methods for the 5th order ODEs (Problems 1 and 2). Problems 1 and 2 were selected to test the 2PVOSBD method in approximating nonlinear ODEs. For Problem 1, the DI method out performed both 1PVOSBD and 2PVOSBD method in terms of accuracy and total steps. The non-homogeneous nature of the problems requires additional steps adjustments which affected the accuracy and total steps of proposed 2PVOSBD. Whereas for Problem 2, the proposed method shows better accuracy for almost every TOL level. It is also shown that in Problem 2, the 2PVOSBD method required less steps as compared to its counterpart when finer TOL is applied.

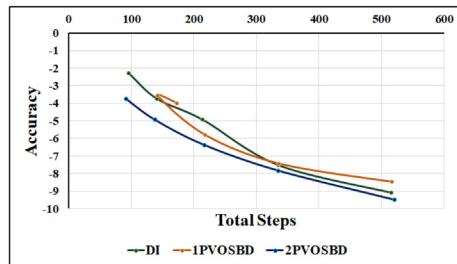
Next, Table 6 compares the numerical results of the 8th (Problem 3) and 6th (Problem 4) orders ODE. These problems were selected due to the order of the problems and large intervals. Prob-



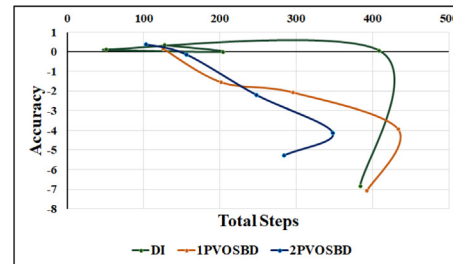
(a) Accuracy over steps for Problem 1.



(b) Accuracy over steps for Problem 2.

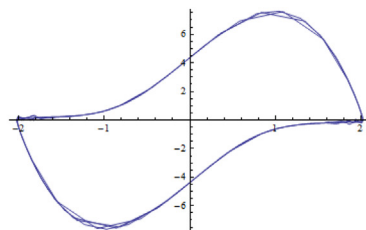


(c) Accuracy over steps for Problem 3.

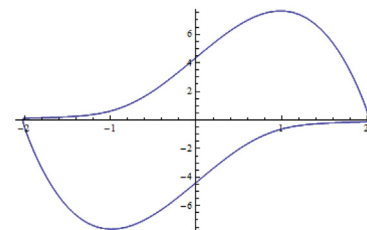


(d) Accuracy over steps for Problem 4.

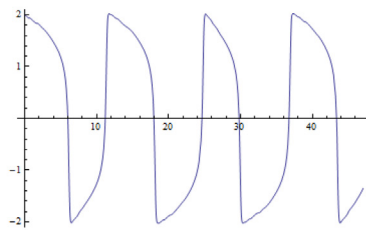
Fig. 2. Efficiency of the DI, 1PVOSBD and 2PVOSBD methods.



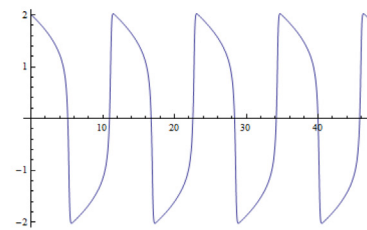
(a) f' vs f using $TOL= 10^{-5}$.



(b) f' vs f using $TOL= 10^{-10}$.



(c) f vs t using $TOL= 10^{-5}$.



(d) f vs t using $TOL= 10^{-10}$.

Fig. 3. Van Der Pol Problem $\mu = 5$ using different TOL levels.

lem 3 is an artificial problem obtained from Suleiman (1979). The accuracy of 2PVOSBD shows to be superior compared to latter methods. The 2PVOSBD out performs the accuracy of the other methods at every TOL. In terms of total steps, the 2PVOSBD and DI method shows to be competitive. Whereas for Problem 4, displays the rivaling nature of accuracy between the 1PVOSBD and 2PVOSBD. To avoid unbiasedness, all methods used for comparison are structured with similar order stepsize conditions. Due to the characteristics of the DI method, certain criteria are not able to be met causing the DI method to be ineffective when dealing with large TOL for this particular problem. This becomes evident at $TOL= 10^{-6}$. Due too many steps failures, the DI is forced to make an early exit from the program.

The accuracy to the total step ratio can be emphasized clearer by using illustrations. The efficiency of Problems 1 to 4 are presented in Fig. 1(a)–(d). To determine their efficiency, the accuracy to the stepsize ratio could be illustrated in the graphical form where the most efficient method is defined in Rasedee (2009) as the undermost curve. Fig. 1 exemplifies the efficiency of the DI for Problem 1(a) whereas the efficiency displayed in Figs. 1(b) and (c) favours the 2PVOSBD. On the other hand, the competitive efficiency between 2PVOSBD and DI is provided in Fig. 1(d), where the DI method is more efficient for large TOL and the 2PVOSBD is more efficient for finer TOL.

Finally, the approximation of Van Der Pol's equation by 2PVOSBD method. Since the Van Der Pol equation is a nonlinear

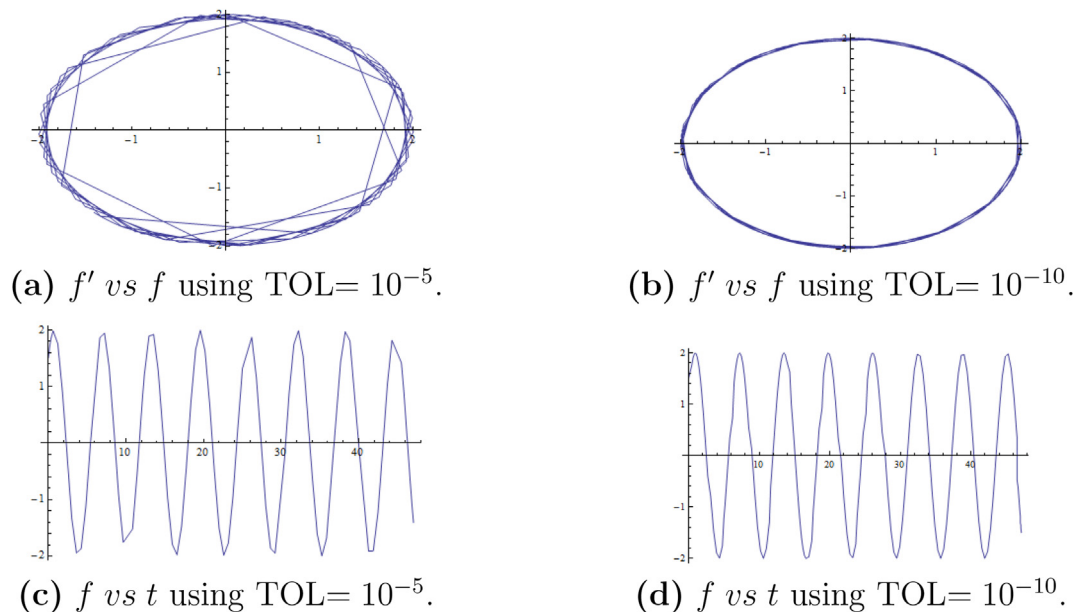


Fig. 4. Van Der Pol Problem $\mu = 0.01$ using different TOL levels.

ODE without any known exact solution, the authors decided that it is best to represent the 2PVOSBD's approximation in graphical form. Figs. 2 and 3 are provided to illustrate the approximation capability of the 2PVOSBD when faced with a non-linear oscillatory equation such as the Van Der Pol's equation. See Fig. 4.

Fig. 2(a)–(d) are the approximated solution for Van Der Pol's equation given $\mu = 5$, $\alpha_1 = 2$, $\alpha_2 = 0$ using two separate TOL, 10^{-5} and 10^{-10} . It can be easily seen that Fig. 2(b) and (d) provide a smoother approximation as compared to Fig. 2(a) and (c). This shows the effectiveness of the 2PVOSBD method when using finer TOL. This becomes evident in Fig. 3(a)–(d) (given $\mu = 0.01$, $\alpha_1 = 1.5$, $\alpha_2 = 1.32288$).

The rapid stepsize increase which is due to a large TOL level, clearly has a more obvious effect in the accuracy of the approximation as shown in Fig. 3(a) and (c). By implementing a finer TOL level, Fig. 3(b) and (d) shows a smoother approximation for the current problem.

In conclusion, the proposed 2PVOSBD method is a viable instrument for solving higher order ODEs. This is contributed to the formulation of its algorithm. With a recursive relationship between different orders, each set of coefficients can be expressed in terms of the previous order. This eliminates the need to reformulate the integration coefficients for different orders but can simply be obtained using the recursive algorithm which reduces computational cost significantly. The efficiency of the algorithm is then elevated with the relationship between the explicit and implicit coefficients which allows the corrector code to be written in terms of the predictor, eliminates the need for tedious to obtain the corrector form scratch.

An extra advantage of the 2PVOSBD method is its adaptability to parallel programming because even with the current accuracy per step, there is an obvious difference between the constant and variable approaches. Problems selected for this research is to challenge the 2PVOSBD method's capabilities. The method has also been tested against the second order systems of ODEs which readers may refer to in the works of Rasedee et al. (2017a, 2018b). The authors are currently working in effort of refine the order and step change for both selection criteria and algorithm to overcome any shortcomings in order to provide a more consistent and optimum approximation for every TOL level selected regardless of the type

of ODE. For future works, the 2PBVOS method can modified for parallel programming and where reduced computational time can be observed. The proposed method can be extended to a three, four and eventually a N block algorithm.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research has been supported by the Ministry of Education Malaysia (MoE) under the Fundamental Research Grant Scheme (FRGS), project number USIM/FRGS/FEM/ 055002/51517 and Universiti Putra Malaysia under the Putra Grant Scheme, project number GP-IPM95859600.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at <https://doi.org/10.1016/j.jksus.2021.101376>.

References

- Al-Jawary, M., Adwan, M., Radhi, G., 2020. Three iterative methods for solving second order nonlinear odes arising in physics. *Journal of King Saud University-Science* 32 (1), 312–323.
- Asnor, A.I., Mohd Yatim, S.A., Ibrahim, Z.B., 2019. Solving directly higher order ordinary differential equations by using variable order block backward differentiation formulae. *Symmetry* 11 (10), 1289.
- Colbrook, M.J., Fokas, T.S., Hashemzadeh, P., 2019. A hybrid analytical-numerical technique for elliptic pdes. *SIAM Journal on Scientific Computing* 41 (2), 1066–1090.
- Collatz, L., 1960. *The numerical treatment of differential equations*.
- Fu, Z.-J., Reutskiy, S., Sun, H.-G., Ma, J., Khan, M.A., 2019. A robust kernel-based solver for variable-order time fractional pdes under 2d/3d irregular domains. *Applied Mathematics Letters* 94, 105–111.
- Gear, C.W., 1967. *The numerical integration of ordinary differential equations*. *Mathematics of Computation* 21 (98), 146–156.
- Goswami, A., Singh, J., Kumar, D., et al., 2019. An efficient analytical approach for fractional equal width equations describing hydro-magnetic waves in

- cold plasma. *Physica A: Statistical Mechanics and its Applications* 524, 563–575.
- Hall, G., Watt, J.M., 1976. *Modern Numerical Methods for Ordinary Differential Equations*. Clarendon Press, Oxford.
- Henrici, P., 1962. *Discrete Variable Methods in Ordinary Differential Equations*. Wiley, New York.
- Ibrahim, Z.B., Zainuddin, N., Othman, K.I., Suleiman, M., Zawawi, I.S.M., 2019. Variable order block method for solving second order ordinary differential equations. *Sains Malaysiana* 48 (8), 1761–1769.
- Khataybeh, S., Hashim, I., Alshbool, M., 2019. Solving directly third-order odes using operational matrices of bernstein polynomials method with applications to fluid flow equations. *Journal of King Saud University-Science* 31 (4), 822–826.
- Krogh, F.T., 1968. A variable-step, variable-order multistep method for the numerical solution of ordinary differential equations. *IFIP Congress* 1, 194–199.
- Krogh, F.T., 1973. Algorithms for changing the step size. *SIAM Journal on Numerical Analysis* 10 (5), 949–965.
- Lambert, J. (1973). *Computational methods in ordinary differential equations*. Chichester, New York..
- Lehrenfeld, C., Olshanskii, M., 2019. An eulerian finite element method for pdes in time-dependent domains. *ESAIM: Mathematical Modelling and Numerical Analysis* 53 (2), 585–614.
- Majid, Z.A., Suleiman, M., 2006. Direct integration implicit variable steps method for solving higher order systems of ordinary differential equations directly. *Sains Malaysiana* 35 (2), 63–68.
- Majid, Z.A., Suleiman, M., 2007. Two point fully implicit block direct integration variable step method for solving higher order system of ordinary differential equations. *World Congress on Engineering*, 812–815.
- Mohd Ijam, H., Ibrahim, Z.B., 2019. Diagonally implicit block backward differentiation formula with optimal stability properties for stiff ordinary differential equations. *Symmetry* 11 (11), 1342.
- Mohd Ijam, H., Ibrahim, Z.B., Abdul Majid, Z., Senu, N., 2020. Stability analysis of a diagonally implicit scheme of block backward differentiation formula for stiff pharmacokinetics models. *Advances in Difference Equations* 2020 (1), 1–22.
- Mohd Ijam, H., Suleiman, M., Rasedee, A.F.N., Senu, N., Ahmadian, A., Salahshour, S., 2014. Solving nonstiff higher-order ordinary differential equations using 2-point block method directly. *Abstract and Applied Analysis* 2014, 1–13.
- Mohd Ijam, H., Ibrahim, Z.B., Senu, N., Suleiman, M., Rasedee, A.F.N. (2018). Order and stability of 2-point block backward difference method. In: *AIP Conference Proceedings*, vol. 1774, AIP Publishing LLC, p. 020054..
- Ola Fatunla, S., 1991. Block methods for second order odes. *International Journal of Computer Mathematics* 41 (1–2), 55–63.
- Omar, Z., Suleiman, M., 2005. Solving higher order ordinary differential equations using parallel 2-point explicit block method. *Matematika* 21 (1), 15–23.
- Rasedee, A., 2009. Direct method using backward difference for solving higher order ordinary differential equations. *University Putra of Malaysia, Selangor*.
- Rasedee, A.F.N., Suleiman, M.b., Ibrahim, Z.B., 2014. Solving nonstiff higher order odes using variable order step size backward difference directly. *Mathematical Problems in Engineering* 2014, 1–11.
- Rasedee, A.F.N. b., Sathar, M.H. b. A., Deraman, F., Ijam, H.M., Suleiman, M. b., Saaludin, N. b., Rakhimov, A. (2016). 2 point block backward difference method for solving riccati type differential problems. In: *AIP Conference Proceedings*, vol. 1775, AIP Publishing LLC, p. 030005..
- Rasedee, A.F.N., Sathar, M.H.A., Wong, T.J., Hamzah, N.A., Koo, L.F., 2021a. Numerical Solution for Duffing-Van Der Pol Oscillator Via Block Method. *Plos One* 10 (1), 19–28.
- Rasedee, A.F.N., Sathar, M.H.A., Wong, T.J., Koo, L.F., Ramli, N.A., 2021. Numerical Solutionf Duffing-Van Der Pol Oscillator Via Block Method. *Advances in Mathematics: Scientific Journal* 10 (1), 19–28.
- Rasedee, A.F.N., Ishak, N., Hamzah, S.R., Ijam, H.M., Suleiman, M., Ibrahim, Z.B., Sathar, M.H.A., Ramli, N.A., Kamaruddin, N.S., 2017a. Variable order variable stepsize algorithm for solving nonlinear duffing oscillator. *Journal of Physics: Conference Series* 890, 012045.
- Rasedee, A.F.N., Sathar, M.H.A., Ishak, N., Kamarudin, N.S., Nazri, M.A., Ramli, N.A., Ismail, I., Sahrim, M., 2017b. Solution for nonlinear duffing oscillator using variable order variable stepsize block method. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics* 33 (2), 165–176.
- Rasedee, A.F.N., Ijam, H.M., Sathar, M.H.A., Ishak, N., Hamzah, S.R., Sahrim, M., Ismail, I. (2018a). Solution for nonlinear riccati equation by block method. In: *AIP Conference Proceedings*, vol. 1774, AIP Publishing LLC, p. 020071..
- Rasedee, A.F.N., Sathar, M.H.A., Ijam, H.M., Othman, K.I., Ishak, N., Hamzah, S.R., 2018b. In: A numerical solution for duffing-van der pol oscillators using a backward difference formulation In *AIP Conference Proceedings*, vol. 2016. AIP Publishing LLC.. 020120.
- Rutishauser, H., 1960. Bemerkungen zur numerischen integration gewöhnlicher differentialgleichungen n-ter ordnung. *Numerische Mathematik* 2 (1), 263–279.
- Singh, H., Kumar, D., Singh, J., Singh, C.S., 2019. A reliable numerical algorithm for the fractional klein-gordon equation. *Engineering Transactions* 67 (1), 21–34.
- Srivastava, H., Dubey, V., Kumar, R., Singh, J., Kumar, D., Baleanu, D., 2020. An efficient computational approach for a fractional-order biological population model with carrying capacity. *Chaos, Solitons & Fractals* 138, 1–13.
- Suleiman, M., 1979. Generalized multistep adams and backward differentiation methods for the solution of stiff and non-stiff ordinary differential equations. Ph. D. thesis, University of Manchester..
- Suleiman, M.B., 1993. Some necessary conditions for convergence of the gbdf methods. *Mathematics of Computation* 60(202) 635–649..
- Suleiman, M., Binti Ibrahim, Z.B., Bin Rasedee, A.F.N., 2011. Solution of higher-order odes using backward difference method. *Mathematical Problems in Engineering*, 2011..
- Veerasha, P., Prakasha, D., Kumar, D., Baleanu, D., Singh, J., 2020. An efficient computational technique for fractional model of generalized hirota-satsuma coupled kdv and coupled mkdv equations. *Journal of Computational and Nonlinear Dynamics* 15 (7), 1–14.
- Waeleh, N. and Abdul Majid, Z., 2016. A 4-point block method for solving higher order ordinary differential equations directly. *International Journal of Mathematics and Mathematical Sciences*, 2016..
- Watts, H.A., Shampine, L., 1972. A-stable block implicit one-step methods. *BIT Numerical Mathematics* 12 (2), 252–266.
- Zainuddin, N., Ibrahim, Z.B., Othman, K.I., Suleiman, M., 2016. Direct fifth order block backward differentiation formulas for solving second order ordinary differential equations. *Chiang Mai J. Sci.* 43, 1171–1181.
- Zhang, H., Qian, X., Song, S., 2020. Novel high-order energy-preserving diagonally implicit runge-kutta schemes for nonlinear hamiltonian odes. *Applied Mathematics Letters* 102, 1–9.