Block Solver for Multidimensional Systems of Ordinary Differential Equations

JIMEVWO GODWIN OGHONYON
Department of Mathematics, Covenant University,
Km 10. Idiroko, Canaan Land,
Ota, Ogun State, NIGERIA
https://covenantuniversity.edu.ng/

SOLOMON ADEWALE OKUNUGA
Department of Mathematics, University of Lagos
Lagos State, NIGERIA
https://unilag.edu.ng

PETER OLUWATOMI OGUNNIYI
Department of Mathematics, Covenant University
Km 10. Idiroko, Canaan Land
Ota, Ogun State, NIGERIA
https://covenantuniversity.edu.ng/

Abstract- This research study aimed at developing block solver for multidimensional systems (BSMS) of ordinary differential equations. This method will be formulated via interpolation and collocation techniques with multinomial as the basis function approximate. The block solver has the capacity to utilize each principal local truncation errors to generate the convergence criteria that will ensure convergence. Some theoretical properties will be stated. The process for executing the block solver will be done via the idea of the convergence criteria introduced. Step by step implementation algorithm will be specified. Some selected model applications will be worked out and a suitable step size will be determined to satisfy the convergence criteria in order to enhance the accuracy and efficiency of the method. The implementation of BSMS is coded in Mathematica and executed under the platform of Mathematica Kernel 9.

Key-words- Block solver; interpolation and collocation; multidimensional systems; variable step size; model applications; convergence criteria, implementation algorithm; Mathematica Kernel

Received: May 23, 2021. Revised: March 21, 2022. Accepted: April 22, 2022. Published: May 24, 2022.

1 Introduction

A large number of physical occurrences are framed with more than one equation and take more than one subordinate variable. For instance, whenever we intend to find out the population of two acting population like foxes and rabbit, we will accept two subordinate variables which constitute the two populations where these populations rely on one autonomous variable which constitutes time. Occurrences such as this give rise to systems of differential equations. See [1] for details.

Consider the initial value problem for a system of r first-order differential equations which possess the general class of

$$Y_1'(x) = f_1(x, Y_1(x), \dots, Y_r(x)),\ Y_1(x_0) = Y_{1,0}, \vdots$$
$$Y_r'(x) = f_1(x, Y_1(x), \dots, Y_r(x)),\ Y_r(x_0) = Y_{r,0}. \quad (1)$$

Seeking for the solution functions $Y_1(x), \dots, Y_r(x)$ on some time-interval defined at $x_0 \le x \le b$.

The general class of (1) is cumbersome to figure out, and it is not easy to determine the system when writing a computer programming. The class of (1) constitute the analytical solution and the differential

equations by employing column vectors. This will be announce as

$$Y(x) = \begin{bmatrix} Y_1(x) \\ \vdots \\ Y_r(x) \end{bmatrix}, \qquad Y_0 = \begin{bmatrix} Y_{1,0} \\ \vdots \\ Y_{r,0} \end{bmatrix},$$

$$f(x,y) = \begin{bmatrix} f_1(x, y_1, \quad \dots \quad y_r) \\ \vdots \\ f_r(x, y_1, \quad \dots \quad y_r) \end{bmatrix} \qquad (2)$$

with $y = [y_1, y_2, \dots, y_r]^T$.
Then (1) will be seen as

$$Y'(x) = f\big(x, Y(x)\big), \qquad Y(x_0) = Y_0. \qquad (3)$$

The notational system of (2) is define as
$f(x,y) = Ay + G(x), \quad y = [y_1, y_2]^T$. [1, 5]

Definition: A system of ordinary differential equations is a concurrent set of equations that takes two or more subordinate variables that rely on one autonomous variable. A solution of the system is a set of functions that meets each equation on some time interval I [1].

Theorem 1: Assume that each of the functions $f_1(x, y_1, y_2, \dots, y_r), f_2(x, y_1, y_2, \dots, y_r),$, $\dots, f_r(x, y_1, y_2, \dots, y_r)$ and the partial derivatives $\frac{\partial f_1}{\partial x_1}, \frac{\partial f_2}{\partial x_2}, \dots, \frac{\partial f_n}{\partial x_n}$ are continuous in a region R establishing the point $(x, y_1, y_2, \dots, y_r)$. Then, the initial-value problem

$$\begin{cases} y_1' = f_1(x, y_1, y_2, \dots, y_r) \\ y_2' = f_2(x, y_1, y_2, \dots, y_r) \\ \vdots \\ y_r' = f_r(x, y_1, y_2, \dots, y_r) \\ y_1(x_0) = t_1, \dots, y_r(x_0) = t_r \end{cases} \qquad (4)$$

has a unique analytical solution

$$\begin{cases} y_1 = \emptyset_1(x) \\ y_2 = \emptyset_2(x) \\ \vdots \\ y_r = \emptyset_r(x) \end{cases} \qquad (5)$$

on the interval I establishing $x = x_0$ [1-2, 15-16].

The system of initial value problems come into existence by nature from any multidimensional system or any system that possess more than one variable quantity associated in a single model equation. Each one of these variable quantities can be constituted by a mathematical function of a single independent variable quantity (usually time) [14].

Theorem 2: Let $f: [0,1] \to \mathbb{R}$ be a continuous function and for each $\mathcal{E} > 0$ there exist a multinomial function P such that for all $x \in [0,1], |f(x) - P(x)| < \varepsilon$. Very importantly, for whatever such f, there exist a succession P_n of multinomial such that $P_n \to f$ uniformly on $[0,1]$. [6]

Authors have suggested possible solutions to handle (1). Some of these methods include the continuous block backward differentiation formula for solving stiff ODEs designed by [2]. This CBBDF requires no starting values and implement implicit block method for solving stiff ODEs. Block method implemented by CBBDF has been well emphasized as an advantage over other methods for ensuring better efficiency and accuracy. Nevertheless, the step size variation and tolerance level were not implemented. The extended continuous block backward differentiation formula for stiff systems is carried out by [3]. This method successfully avoids the use of starting values and thereby executed block method approach with fixed step size. [3] utilizes the fixed step approach to determine the numerical result. This approach of fixed step size is not comparable with variable step size change and tolerance level approach which guarantees convergence of every iteration. Block hybrid k-step backward differentiation formulas for large stiff systems has be executed by [13]. This method possesses the properties of block backward differentiation formula and has the vantage of avoiding the use of predictor method for initializing the process. [13] solved both linear stiff and non stiff systems utilizing fixed step size without implementing step size change and tolerance level to guarantee better efficiency and accuracy. [17] implemented the numerical solution of first order stiff ODEs using fifth order block backward differentiation formulas. The idea is basically for stiff ODEs which solved large systems of ODEs simultaneously with fixed step size as against finding a suitable step size and including tolerance level. The parallel implementation of the parallel block backward differentiation formulas displays important benefits above the successive implementation. The idea of step size change and tolerance level was implemented but the technique of finding a suitable step size for each tolerance level was not examined by [23]. The derivation of block solver for multidimensional systems is the main aim of this research study. The introduction of block solver for multidimensional systems (BSMS) of ODEs originates as a result of the bounded stability attributes of the block solver which contributes to the unfitness of the system to show better efficiency and accuracy. Again, BSMS is proposed by [15-16] to outwit the Dahlquist

roadblocks. Again, [15-16] suggested the introduction of variable step variable and variable step size as an option to better accuracy and efficiency. BSMS will ensure better efficiency and accuracy by the introduction of variable step variable order and finding a suitable variable step size to ensure the satisfaction of the convergence criteria. BSMS possesses the idea of block method and the implementation process is done by a specially designed formula to achieve the desired result thereby ensuring error control or monitoring.

The contribution of this research study is the derivation and implementation of a block solver for multidimensional systems via variable step variable order and finding a suitable step size. Block solver as a result of the multidimensional systems will be implemented like other block backward differentiation formulas adopting block method approach.

2 Developing the Block Solver of Multidimensional Systems

The process of developing the block solver for multidimensional systems of ODEs (BSMS) will be done via interpolation and collocation together with multinomial as the basis function approximate. This propose block solver is constructed using $k+1-step$ of block predictor mode of order $p+1$ while the block corrector mode uses $k-step$ of order p. This combination is referred to as the block solver defined in form of variable step and variable order.

The block predictor mode utilizes $y_{n-i},\ i=0$ as the interpolation point and $f_{n-i},\ i=0,1,2,3$ as the collocation points. On the other hand, the block corrector mode takes on $y_{n-i},\ i=2$ for interpolation and $f_{n+i},\ i=1,2,3$ as the points of collocation. The derivation of block solver is subjected to a special multinomial basis function approximate

$$y(x)=\sum_{i=0}^{j} a_i\left(\frac{x-x_n}{h}\right)^i, \tag{6}$$

where $a_i,\ i=0,1,2,3,4$ constitute the unknown physical quantities required to be examine specially. Whenever (6) is utilized to approximate (1), theorem 2 is satisfied. Hence, the analytical solution of the points of interpolation $x=x_{n-i},\ i=0\ and\ x=x_{n-i},\ i=2$ will produce

$$y(x_n)\approx y_n,\quad y(x_{n-i})\approx y_{n-2}, \tag{7}$$

and points of collocation, $x=x_{n-i},\ i=0,1,2,3,\ x_{n+i},\ i=1,2,3$ to bring forth

$$y'(x_{n-i})\approx f_{n-i},\ i=0,1\,2,3,\ y'(x_{n+i})\approx f_{n+i},\ i=0,1\,2,3. \tag{8}$$

Equations (7) and (8) will be combined together to produce the system of equations in the form of $AX=B$. The solution loop will converge whenever the absolute values of the pre-eminent diagonal components of the constant coefficient square matrix A of the system $AX=B$ are larger than the total of absolute values of the other constant coefficient of the row [11]. Calculating $AX=B$ and after then, substituting the results into (6) as well as evaluating at some selected interval of $x=x_{n+i},\ i=1,2,3$ will give rise to block predictor mode and block corrector mode of

$$\begin{aligned} y_{n+1}&=\sum_{i=0}^{1}\alpha_i y_{n-i}=h\sum_{i=0}^{3}\beta_i f_{n-i}\\ y_{n+1}&=\sum_{i=2}^{1}\alpha_i y_{n-i}=h\sum_{i=1}^{3}\beta_i f_{n+i}\end{aligned} \tag{9}$$

Equation (9) is called block solver for multidimensional systems of ODEs [18-22]. Block solver developed by (9) has variable step and variable order. The block predictor mode has 4-step of order 4 while the block corrector mode has 3-step of order 3. This combination is a special design of a higher block predictor mode with a lower block corrector mode.

2.1 Theoretical Properties of the Method

Theorem 3: Whenever the block solver (9) converges to a certain pth order of equations then the order of (9) is at leastwise p [9].

Theorem 4: The order of (9) for first order equations must be greater than or equal to one whenever it is convergent [9].See [9] for proof.

2.2 Executing the Convergence Criteria of Block Solver

The usage of block solver for estimating the principal local truncation error call for the block predictor mode-block corrector mode to own ilk order. To realized this we allow the block predictor to be $q-step$ Adams Bashforth method and block corrector to be $(q-1)-step$ Adams-Moulton method, both then own $p=k$. The $q-step\ with\ kth$ order ABM pair is therefore

$$\left.\begin{aligned} y_{n+1} &= y_n + h\sum_{i=0}^{q-1}\gamma_i^*\nabla^i f_n,\ p^* = k, \qquad C_{k+1}^* = \gamma_k^* \\ y_{n+1} &= y_n + h\sum_{i=0}^{q-1}\gamma_i\nabla^i f_{n+1},\ p = k, \qquad C_{k+1} = \gamma_k \\ k &= 1,2,\ldots \end{aligned}\right\} \quad (10)$$

Whenever we presume (10) to be utilized in $P(EC)^{\mu}E^{1-t}$ mode, then, in the second of (10), y_{n+1} will be replaced by $y_{n+1}^{[v+1]}$, and the value f_{n+1} displayed on the right side by $f_{n+1}^{[v]}$, the remaining values of f_{n-q} being replaced by $f_{n-q}^{[\mu-t]}, q = 0,1,\ldots,k-1$. We can surmount this task by specifying $\nabla_v^i f_{n+1}^{[\mu]}$ as $\nabla^i f_{n+1}^{[\mu]}$ with the one value $f_{n+1}^{[\mu]}$ replaced as $f_{n+1}^{[v]}$. That is,

$$\nabla_v^i f_{n+1}^{[\mu]} = \nabla^i f_{n+1}^{[\mu]} + f_{n+1}^{[v]} - f_{n+1}^{[\mu]} \quad (11)$$

We may rewrite (30) in the class

$$\sum_{i=0}^{k-1}\left(\gamma_i\nabla^i f_{n+1}^{[\mu]} - \gamma_i^*\nabla^i f_n^{[\mu]}\right) = \gamma_{k-1}^* f_{n+1}^{[\mu]} \quad (12)$$

and represent (10) as the notational system.

Right away, the pair of (10) is utilized in $P(EC)^{\mu}E^{1-t}$ mode, and employs the mode of Adams methods to construct a type of ABM method that is very tedious and difficult to handle in terms of the computation. The type is defined as follows:

$$P: \quad y_{n+1}^{[0]} = y_n^{[\mu]} + h\sum_{i=0}^{k-1}\gamma_i^*\nabla^i f_n^{[\mu-1]} \quad (13)$$

$(EC)^{\mu}$

$$\left.\begin{aligned} f_{n+1}^{[v]} &= f\left(x_{n+1}, y_{n+1}^{[v]}\right) \\ y_{n+1}^{[v+1]} &= y_n^{[\mu]} + h\sum_{i=0}^{k-1}\gamma_i\nabla_v^i f_{n+1}^{[\mu-1]} \end{aligned}\right\} \ v = 0,1,\ldots,\mu-1 \quad (14)$$

$(E^{1-t}$

$$f_{n+1}^{[\mu]} = f\left(x_{n+1}, y_{n+1}^{[\mu]}\right)$$

whenever $t = 0$.

To utilize the block solver, we demand the calculation of $y_{n+1}^{[\mu]} - y_{n+1}^{[0]}$. Deducting (13) from (14) with $v = \mu - 1$ we have

$$y_{n+1}^{[\mu]} - y_{n+1}^{[0]} = h\sum_{i=0}^{k-1}\left(\gamma_i\nabla_{\mu-1}^i f_{n+1}^{[\mu-t]} - \gamma_i^*\nabla^i f_n^{[\mu-t]}\right) = h\gamma_i^*\nabla_{\mu-1}^k f_{n+1}^{[\mu-t]}.$$

Since $C_{k+1}^* = \gamma_k^*$ and $C_{k+1} = \gamma_k$, the block solver estimate will be seen as

$$W = \frac{C_{p+1}}{C_{p+1}^* - C_{p+1}} < \boldsymbol{\varepsilon}. \quad (15)$$

Whenever the principal local truncation error is at x_{n+1}, the T_{n+1} is achieved as

$$\left.\begin{aligned} T_{n+1} &= \frac{C_{p+1}}{C_{p+1}^* - C_{p+1}}\left(y_{n+1}^{[\mu]} - y_{n+1}^{[0]}\right) = \frac{\gamma_k}{\gamma_k^* - \gamma_k} = \\ & h\gamma_i^*\nabla_{\mu-1}^k f_{n+1}^{[\mu-t]}. \end{aligned}\right\},$$

Whenever $\gamma_k^* - \gamma_k = \gamma_{k-1}^*$, wherefrom $T_{n+1} == h\gamma_i^*\nabla_{\mu-1}^k f_{n+1}^{[\mu-t]}$. See [4, 7, 15-16, 18-22] for more info.

2.3 Step by Step Implementation Algorithm of Block Solver

Step 1: choose a step size h and vary the step size until a suitable variable step size h is found

Step 2: use Taylor's series of order four to prime the block solver.

Step 3: write the code of block solver using Mathematica

Step 4: run equation (13) with step 1 under the platform of Mathematica Kernel 9.

Step 5: if step 4 fails repeat the process again as prescribed by step 1.

Step 6: if step 5 is successful after determining the suitable variable step size h then proceed to step 7.

Step 7: print the maximum errors of the block solver.

3 Results and Discussion

Three model applications will be examine to show case better efficiency and accuracy of the block solver. The computational results of BSMS be compared with the analytical result and verified using some selected convergence criteria of $\mathbf{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}}$ ***and*** $\mathbf{10^{-8}}$. The BSMS is applied under a proficient mode in the manner of $\boldsymbol{P(EC)^{u}}$ to examine the convergence criteria, efficiency, accuracy and maximum error. The block solver of (9) will be devised and carried out under the platform of Mathematica to solve the model applications of the multidimensional systems of ODEs.

3.1 Numerical Examples

We utilized the idea of compartment analysis to transforms the diagram into a system of linear differential equations. The concept has been utilized to formulate real life modes in various topics such as environment science, chemical science, heating, cooling, kinetics, mechanics and electrical energy.

A compartment diagram consists of the following elements.

Variable Names: Each compartment is marked with a variable X.

Arrows: Each arrow is labeled with flow rat R.
Input Rate: An arrowhead directing at compartment X documents input rate R.
Output Rate: An arrowhead directing away from compartment X documents output rate R.

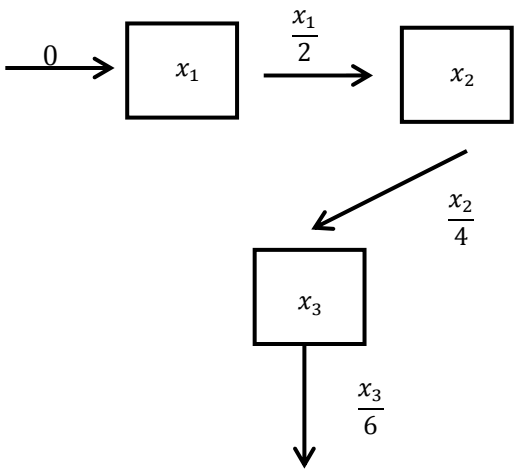

Fig. 1: A diagram of compartment analysis. The diagram represents the classical brine tank problem of figure 2.

Gathering the single linear differential equation for a diagram compartment X is carried out by composing $\frac{dX}{dt}$ for the left hand side of the differential equation. Again, in algebraic manner we add the input and output rates to get the right hand side of the differential equation, allowing to the equilibrium law.

$$\frac{dX}{dt} = sum\ of\ input\ rates - sum\ of\ output\ rates$$

Conventionally, a compartment that has no coming arrowhead possess input zero, and a compartment that has no coming out possess output zero.

These model applications of the systems of ODEs are as follows

- Recycled brine tank cascade [10]
- Biomass Transfer [10].
- Population problems [1].

Model Application 1: Three brink tanks in cascade with recycling

Let brine tanks A, B, C be given volumes of $\mathbf{60}, \mathbf{30}, \mathbf{60}$, respectively, as in figure 2

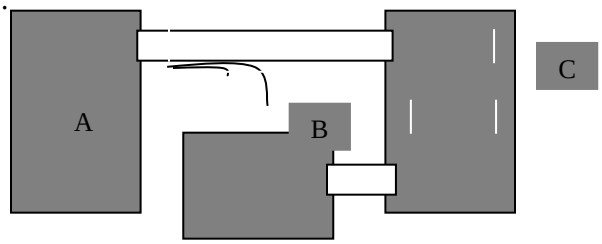

Fig. 2: Three brine tanks in cascade with recycling.

Suppose that fluid drains from tank A to B at rate r, drains from tank B to C at rate r, then drains from tank C to A at rate r. The tank volumes remain constant due to constant recycling of fluid. For the purpose of illustration, let $\boldsymbol{r = 10}$. Uniform stirring of each tank is assumed, which implies uniform salt concentration throughout each tank.

Let $x(t), y(t), z(t)$ denote the amount of salt at time t in each tank. No salt is lost from the system, due to recycling. Using compartment analysis, the recycled cascade is modeled by the non-triangular system

$$x'(t) = -\frac{1}{6}x + \frac{1}{6}z,$$
$$y'(t) = \frac{1}{6}x - \frac{1}{3}y,$$
$$z'(t) = \frac{1}{3}y - \frac{1}{6}z.$$

The analytical solution is given by

$$x(t) = 2 - \mathrm{e}^{-\frac{t}{3}}\cos\left(\frac{t}{6}\right) + 3\mathrm{e}^{-\frac{t}{3}}\sin\left(\frac{t}{6}\right),$$
$$y(t) = 1 - \mathrm{e}^{-\frac{t}{3}}\sin\left(\frac{t}{6}\right) - 3\mathrm{e}^{-\frac{t}{3}}\cos\left(\frac{t}{6}\right),$$
$$z(t) = 2 - 2\mathrm{e}^{-\frac{t}{3}}\sin\left(\frac{t}{6}\right) + 4\mathrm{e}^{-\frac{t}{3}}\cos\left(\frac{t}{6}\right).$$

At infinity, $x = z = c_1,\ y = \frac{c_1}{2}$. This implies that the total amount of salt is uniformly distributed in the tanks, in the ratio $2:1:2$ [10].

Model Application 2: Biomass Transfer

Consider a European forest having one or two varieties of trees. We select some of the oldest trees, those expected to die off in the next few years, and then follow the cycle of living trees into dead trees. The dead trees eventually decay and fall from seasonal and biological events. Finally, the fallen trees become humus. Let variables x, y, z, t be

$x(\mathrm{t}) = \text{biomass decayed into humus},$
$y(\mathrm{t}) = \text{biomass of dead trees},$
$z(\mathrm{t}) = \text{biomass of living trees},$
$\mathrm{t} = \text{time in decades (decade} = 10\text{years)}.$

A typical biological model is

$$x'(t) = -x(t) + 3y(t),$$
$$y'(t) = -3y(t) + 5z(t),$$

$z'(t) = -5z(t)$.

Suppose there are no dead trees and no humus at $t = 0$, with initially z_0 units of living tree biomass. These assumptions imply initial conditions $x(0) = y(0) = 0, z(0) = z_0$. The analytical solution is seen as

$x(\mathrm{t}) = \frac{15}{8}\mathrm{z}_0(\mathrm{e}^{-5\mathrm{t}} - 2\mathrm{e}^{-3\mathrm{t}} + \mathrm{e}^{-\mathrm{t}})$,

$y(\mathrm{t}) = \frac{5}{2}\mathrm{z}_0(\mathrm{e}^{-5\mathrm{t}} + \mathrm{e}^{-3\mathrm{t}})$,

$z(t) = z_0 e^{-5t}$.

The live tree biomass $z(t) = z_0 e^{-5t}$ decreases according to a Malthusian decay lay from its initial size z_0. It decays to 60% of its original biomass in one year [10].

Model Application 3: Population problem

The rate at which population $x, y, and\ z$ changes is

$\frac{dx}{dt} = (a_1 - a_2 - a_3)x + b_2 y + c_2 z$,

$\frac{dy}{dt} = (b_1 - b_2 - b_3)y + a_2 x + c_3 z$

$\frac{dz}{dt} = (c_1 - c_2 - c_3)z + a_3 x + b_3 y$,

where the initial population $x(0) = 50, y(0) = 60\ and\ z(0) = 25$ while $(a_1, a_2, a_3) = (3, 0, 2)$, $(b_1, b_2, b_3) = (4, 2, 1)$ and $(c_1, c_2, c_3) = (5, 3, 0)$ are given.

The exact solutions are

$x(t) = e^{-t}(-3 - 10e^{2t} + 63e^{5t})$,

$y(t) = 60e^{t}$,

$z(t) = e^{-t}(2 - 40e^{2t} + 63e^{5t})$ [1].

TERMINOLOGY

BSMS : block solver for multidimensional systems of ODEs.

$\boldsymbol{\varepsilon}$: tolerance level of the convergence criteria.

$MAXE$: maximal error(s).

Table 1. Results of Model Application 1

Method Used	MAXE	Convergence Criteria
BSMS	2.18648×10^{-7}	10^{-4}
BSMS	2.27731×10^{-7}	
BSMS	2.33203×10^{-8}	
BSMS	3.43747×10^{-12}	10^{-6}
BSMS	2.4776×10^{-11}	
BSMS	2.21512×10^{-12}	
BSMS	4.44089×10^{-16}	10^{-8}
BSMS	2.66454×10^{-15}	
BSMS	0.	

Table 2. Results of Model Application 2

Method Used	MAXE	Convergence Criteria
BSMS	9.85964×10^{-4}	10^{-3}
BSMS	1.57492×10^{-4}	
BSMS	7.37505×10^{-5}	
BSMS	1.4094×10^{-8}	10^{-5}
BSMS	1.40426×10^{-9}	
BSMS	1.0126×10^{-8}	
BSMS	1.46215×10^{-12}	10^{-7}
BSMS	1.4293×10^{-13}	
BSMS	1.04594×10^{-12}	

Table 3. Results of Model Application 3

Method Used	MAXE	Convergence Criteria
BSMS	4.66481×10^{-7}	10^{-4}
BSMS	3.59918×10^{-7}	
BSMS	4.65102×10^{-7}	
BSMS	4.54676×10^{-11}	10^{-6}
BSMS	3.53992×10^{-11}	
BSMS	4.53291×10^{-11}	
BSMS	1.42109×10^{-14}	10^{-8}
BSMS	7.10543×10^{-15}	
BSMS	7.10543×10^{-15}	

4 Conclusion

The block solver for multidimensional systems (BSMS) of ODEs has been suggested. The block solver is product of block predictor mode and block corrector mode which is formulated utilizing variable step and variable order. Three model applications with exponentially and trigonometrically solutions in nature were examined. The block solver adopted the idea of variable step-variable order and variable step size to implement the procedure. The convergence criteria apply (15) together with (9) to ensure the implementation The convergence criteria of $10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\ and\ 10^{-8}$ were utilized to decide the MAXE results. The mathematical

expression of (15) is used to decide whether to accept or reject the results. The high level of efficiency and accuracy achieved were made possible with the determination of a suitable step size and block solver designed via variable step and variable order. Block solver derived has the capacity to proffer solution to multidimensional systems of ODEs with oscillating and vibration behavior via the efficient utilization of variable step size-variable order and suitable variable step size. A step by step approach for realizing the result is specified. The execution of block solver is implemented under the Mathematica Kernel 9. Thus, makes it easier to achieve faster computation and precise results. Furtherwork is required to build a block solver to handle stiff oscillating and vibration solutions.

Acknowledgements:
The authors would like to appreciate Covenant University, Ota for their continuous sponsorship throughout this research project.

References:

[1] M. L. Abell, J. P. Braselton, *Differential equations with Mathematica*, Elsevier Academic Press, USA, 2004.

[2] O. A. Akinfenwa, S. N. Jator, N. M. Yao, Continuous block backward differentiation formula for solving stiff ordinary differential equations, *Computers and Mathematics with applications*, Vol. 65, No. 7, 2013, pp. 996-1005.

[3] O. A. Akinfenwa, S. N. Jator, Extended continuous block backward differentiation formula for stiff systems, *Fasciculi Mathematici*, Vol. 2015, 2015, pp. 1-15.

[4] U. M. Ascher, L. R. Petzoid, *Computer methods for ordinary differential equations and differential-algebraic equations*, SIAM, USA, 1998.

[5] K. Atkinson, W. Han, D. Stewart, *Numerical solution of ordinary differential equations*, John Wiley & Sons, Inc, New Jersey, 2009.

[6] M. Bond, *Convolutions and the Weierstrass approximation theorem*, Department of Mathematics, Michigan State University, USA, 2009.

[7] J. R. Dormand, *Numerical methods for differential equations*, CRC Press, New York, 1996.

[8] S. O. Fatunla, *Numerical methods for initial value problems in ordinary differential equations*, Academic Press, Inc, New York, 1988.

[9] C. W. Gear, *Numerical value problems in ODEs*, Prentice-Hall, Inc;, New Jersey, USA, 1971.

[10] 'Systems of ODEs', available at http://www.math.utah.edu/~gustafso/2250systems-de.pdf.

[11] S. R. K. Iyengar, R. K. Jain, *Numerical methods*, New Age International (P) Ltd, New Delhi, 2009.

[12] M. K. Jain, S. R. K. Iyengar, R. K. Jain, *Numerical methods for scientific and Engineering computation*, New Age International (P) Limited, New Delhi, India, 2007.

[13] S. N. Jator, E. Agyingi, Block hybrid k-step backward differentiation formulas for large stiff systems, International Journal of Computational Mathematics, Vol. 2014, 2014, 1-9.

[14] R. Khoury, H. Wilhelm, *Numerical methods and modelling for engineering*, Springer Nature, Switzerland, 2016.

[15] J. D. Lambert, *Computational methods in ordinary differential equations*, John Wiley & Sons, New York, 1973.

[16] J. D. Lambert, *Numerical methods for ordinary differential systems*, John Wiley & Sons, New York, 1991.

[17] A. A. M. N. Nor, B. I. Zarina, I. O. Khairil, S. Mohamed, Numerical solution of first order stiff ordinary differential equations using fifth order block backward differentiation formulas, *Sains Malaysiana*, Vol. 41, No. 4, 2012, pp. 489-492.

[18] J. G. Oghonyon, S. A. Okunuga, N. A. Omoregbe, O. O. Agboola, A computational approach in estimating the amount of pond and determining the long time behavioural representation of pond pollution, *Global Journal of Pure and Applied Mathematics*, Vol. 11, No 5, 2015, pp. 2773-2785.

[19] J. G. Oghonyon, J. Ehigie, S. K. Eke, Investigating the convergence of some selected properties on block predictor-corrector

methods and it's applications, *Journal of Engineering and Applied Sciences,* Vol. 11, No 11, 2402-2408, 2016.

[20] J. G. Oghonyon, O. A. Adesanya, H. Akewe, H. I. Okagbue, "Softcode of multi-processing Milne's device for estimating first-order ordinary differential equations, *Asian Journal of Scientific Research*, Vol. 11, No. 4, 2018, pp. 553-559.

[21] J. G. Oghonyon, O. F. Imaga, P. O. Ogunniyi, The reversed estimation of variable step size implementation for solving nonstiff ordinary differential equations, *International Journal of Civil Engineering and Technology*, Vol. 9, No. 8, 2018, pp. 332-340.

[22] J. G. Oghonyon, S. A. Okunuga, H. I. Okagbue, Expanded trigonometrically matched block variable-step-size technics for computing oscillating vibrations, *Proceedings of the International Multi-Conference of Engineerings and Computer Scientists,* Vol. 2019, 2019, pp. 552-557.

[23] B. Zarina, I. Khairil, O. Iskandar, Parallel block backward differentiation formulas for solving large systems of ordinary differential equations, *World Academic of Science, Engineering and Technology*, Vol. 4, No. 4, 2010, 1-4.

Conflicts of Interest

There is no conflict of interest among the authors.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

-Jimevwo Godwin Oghonyon develop the idea, method and write the code using Mathematica.
-Solomon Adewale Okunuga supervised the research work.
-Peter Oluwatomi Ogunniyi provided the logistics and technical support.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

This research study is funded by Covenant University, Ota.