International Journal of Pure and Applied Mathematics
Volume 80 No. 4 2012, 543-559
ISSN: 1311-8080 (printed version)
url: http://www.ijpam.eu
ijpam.eu

BLOCK HYBRID-SECOND DERIVATIVE METHOD FOR STIFF SYSTEMS

F.F. Ngwane[1 §], S.N. Jator[2]

[1]Department of Mathematics
USC Salkehatchie, SC, USA
[2]Department of Mathematics and Statistics
Austin Peay State University
Clarksville, TN 37044, USA

Abstract: A continuous representation based on the Hybrid Second Derivative Method (HSDM) is constructed and used to generate Initial Value Methods (IVMs) for stiff systems of first order ordinary differential equations. The IVMs are applied as simultaneous numerical integrators by assembling them into a single block matrix equation which is A-stable. Numerical results produced by the block method show that the method is competitive with existing ones in the literature.

AMS Subject Classification: 65L05, 65L06
Key Words: first order system, hybrid method, off-step point, block method, second derivative

1. Introduction

In this paper, we consider the first order differential equation

$$y' = f(x, y),\ y(a) = y_0\ ,\ x \ \epsilon\ [a, b] \tag{1}$$

where $f : \Re \times \Re^m \rightarrow \Re^m$, $y, y_0\ \epsilon\ \Re^m$, f satisfies a Lipschitz condition (see Henrici [14]), and the Jacobian $(\frac{\partial f}{\partial y})$, whose eigenvalues have negative real parts,

Received: June 28, 2012

§Correspondence author

varies slowly ([15]). According to Dahlquist [5], the order of a k-step Linear Multistep Method (LMM) cannot exceed $k+1$ (k is odd) or $k+2$ (k is even) for the method to be zero-stable. The Dahlquist barrier theorem was overcome by several authors who proposed modified forms of LMMs known as hybrid methods by incorporating off-step points in the derivation process (see Gear [9], Gragg and Stetter [11], Butcher [3], Gupta [12], Lambert [22], and Kohfeld and Thompson [20]). These methods were shown to be of order up to $2k+2$, but included additional off-grid functions. Gupta [12] noted that the design of algorithms for hybrid methods is more tedious due to the occurrence of off-step functions which increase the number of predictors needed to implement the methods. Second derivative methods which were shown to be of order up to $k+2$ were proposed by Enright [6] and implemented in a variable order, variable-step mode. Exponentially fitted methods have also been proposed for stiff systems of the form (1) (see [4], [15]).

In this paper, we construct a continuous HSDM through interpolation and collocation (see Lie and Norsett [23], Atkinson [1], Onumanyi et al [26], and Gladwell and Sayers [10]). The continuous representation generates a main discrete HSDM and one additional method which are combined and used as a block method to simultaneously produce approximations $\{y_{n+\frac{1}{2}}, y_{n+1}\}$ at a block of points $\{x_{n+\frac{1}{2}}, x_{n+1}\}$, $h = x_{n+1} - x_n$, $n = 0, \dots, N-1$ on a partition $[a, b]$, where $a, b \in \mathbb{R}$, h is the constant step-size, n is a grid index and $N > 0$ is the number of steps.

The method preserves the Runge-kutta traditional advantage of being self-starting and is more accurate since it is implemented as a block method. We note that block methods were first introduced by Milne [25] for the purpose of obtaining starting values for predictor-corrector algorithms (see Sarafyan [29]). However, Rosser [28], developed Milne's idea into algorithms for general use. Block methods have also been considered by Shampine and Watts [30]. We emphasize that our HSDM is developed for general use, not only as a means of obtaining starting values for predictor-corrector algorithms. We emphasize that the continuous representation generates a main discrete HSDM and one additional method which are combined and implemented as a block method which simultaneously generates approximations $\{y_{n+\frac{1}{2}}, y_{n+1}\}$ to the exact solutions $\{y(x_{n+\frac{1}{2}}), y(x_{n+1})\}$.

In order to apply the block method at the next block to obtain y_{n+2}, the only necessary starting value is y_{n+1}, and the loss of accuracy in y_{n+1} does not affect subsequent points, thus the order of the algorithm is maintained. It is unnecessary to make a function evaluation at the initial part of the new block. Thus, at all blocks except the first, the first function evaluation is already available from the previous block. Hence, as we proceed we have four function evaluations per step as in the conventional fourth order Runge-kutta method in spite of the higher order of our method. The method is also more efficient than that given Jator[17] since only two equations are solved per block compared to four equations per block in [17].

The paper is organized as follows. In Section 2, we obtain a continuous representation $U(x)$ for the exact solution $y(x)$ which is used to generates a main discrete HSDM and one additional method for solving (1). The analysis and implementation of the methods are discussed in Section 3. Numerical examples are given in Section 4 to show the efficiency of the methods. Finally, the conclusion of the paper is discussed in Section 5.

2. Development of Method

In this section, our objective is to derive the main HSDM of the form

$$y_{n+1} = y_n + h(\beta_0 f_n + \beta_1 f_{n+1} + \beta_\nu f_{n+\nu}) + h^2(\gamma_0 g_n + \gamma_1 g_{n+1} + \gamma_\nu g_{n+\nu}) \quad (2)$$

and the additional method

$$y_{n+\nu} = y_n + h(\hat{\beta}_0 f_n + \hat{\beta}_1 f_{n+1} + \hat{\beta}_\nu f_{n+\nu}) + h^2(\hat{\gamma}_0 g_n + \hat{\gamma}_1 g_{n+1} + \hat{\gamma}_\nu g_{n+\nu}) \quad (3)$$

where β_j, γ_j, β_ν, γ_ν, and $\hat{\beta}_j$, $\hat{\gamma}_j$, $\hat{\beta}_\nu$, $\hat{\gamma}_\nu$ $j = 0, 1$, are constants and ν is chosen from the interval $(0, 1)$. We note that $y_{n+j\nu}$ is the numerical approximation to the analytical solution $y(x_{n+j\nu})$, $f_{n+j\nu} = f(x_{n+j\nu}, y_{n+j\nu})$, and

$$g_{n+j\nu} = \frac{df(x, y(x))}{dx}\Big|^{x_{n+j\nu}}_{y_{n+j\nu}},$$

$j = 0, 1, 2$.

In order to obtain (2) and (3) on every interval $[x_n, x_n+h]$, $n = 0, 1, \ldots, N-1$, we proceed by seeking to approximate the exact solution $y(x)$ by the interpolating function $U(x)$ of the form

$$U(x) = \sum_{j=0}^{r+2s-1} \ell_j \psi_j(x) \quad (4)$$

where ℓ_j are unknown coefficients and $\psi_j(x)$ are polynomial basis functions of degree $r + 2s - 1$, where the number of interpolation points r and the number of distinct collocation points s are chosen to satisfy $1 \leq r \leq k$, and $s > 0$ respectively. The positive integer $k \geq 1$ denotes the step number of the method.

Letting $\psi_j(x) = x^j$, $j = 0, 1, \ldots, r+2s-1$, we impose that the interpolating function (4) coincides with the analytical solution at the end point x_n $(r = 1)$ to obtain the equation

$$U(x_n) = y_n \tag{5}$$

If the function (4) satisfies the differential equation (1) at the points $x_{n+j\nu}, j = 0, 1, 2$ $(s = 3)$, we obtain the following set of three equations:

$$U^{\prime}(x_{n+j\nu}) = f_{n+j\nu}, j = 0, 1, 2. \tag{6}$$

We further demand that the second derivative of the function (4) coincides with the second derivative of the analytical solution at the points $x_{n+j\nu}, j = 0, 1, 2$ $(s = 3)$ to obtain the following set of three equations:

$$U^{\prime\prime}(x_{n+j\nu}) = g_{n+j\nu}, j = 0, 1, 2. \tag{7}$$

We emphasize that equations (5), (6) and (7) lead to a system of seven equations which is solved by Cramer's Rule to obtain ℓ_j. Our continuous HSDM is constructed by substituting the values of ℓ_j into equation (4). After some manipulation, our method is expressed in the form

$$U(x) = y_n + h(\beta_0(x)f_n + \beta_1(x)f_{n+1} + \beta_\nu(x)f_{n+\nu}) + h^2(\gamma_0(x)g_n + \gamma_1(x)g_{n+1} + \gamma_\nu(x)g_{n+\nu}) \tag{8}$$

where $\beta_0(x)$, $\beta_1(x)$, $\gamma_0(x)$, $\gamma_1(x)$, $\beta_\nu(x)$, and $\gamma_\nu(x)$ are continuous coefficients. The continuous method (8) is used to generate the main HSDM of the form (2) and an additional method of the form (3) by choosing, $r = 1$, $s = 3$, $\nu = \frac{1}{2}$, and $\psi_j(x) = x^j$, $j = 0, 1, \ldots, 6$. Thus, evaluating (8) at $x = \{x_{n+1}, x_{n+\frac{1}{2}}\}$, we generate the following main method and additional method, which are particular methods of the form (2) and (3).

$$y_{n+1} = y_n + \frac{h}{30}(7f_n + 16f_{n+1/2} + 7f_{n+1}) + \frac{h^2}{60}(g_n - g_{n+1}) \tag{9}$$

$$y_{n+1/2} = y_n + \frac{h}{480}(101f_n + 128f_{n+1/2} + 11f_{n+1}) + \frac{h^2}{960}(13g_n - 40g_{n+1/2} - 3g_{n+1}).$$

In particular, (9) and (10) are combined and implemented as a block method which simultaneously generates approximations $y_{n+j/2}$ to the exact solutions $y(x_{n+j/2})$ for $j = 1, 2$ on the partition π_N, where

$$\pi_N : a = x_0 < x_1 < x_2 < ... < x_n < x_{n+1} < \ldots < x_N = b.$$

3. Analysis and Implementation

In this section, we discuss the local truncation error and order, zero-stability, linear-stability, and implementation of the method.

Local truncation error and order. Following Fatunla [7] and Lambert [21] we define the local truncation error associated with (2) to be the linear difference operator

$$L[y(x); h] = \sum_{j=0}^{1}\{\alpha_j y(x + jh) - h\beta_j y'(x + jh) - h^2\gamma_j y''(x + jh)\} - h\beta_\nu y'(x + \nu h) - h^2\gamma_\nu y''(x + \nu h). \quad (10)$$

Assuming that $y(x)$ is sufficiently differentiable, we can expand the terms in (11) as a Taylor series about the point x to obtain the expression

$$L[y(x); h] = C_0 y(x) + C_1 h y'(x) + \ldots + C_q h^q y^{(q)}(x) + \ldots \quad (11)$$

where the constant coefficients C_q , $q = 0, 1, \ldots$ are given as follows:

$$C_0 = \sum_{j=0}^{1} \alpha_j,$$

$$C_1 = \sum_{j=1}^{1} j\alpha_j - \sum_{j=0}^{1} \beta_j - \beta_\nu,$$

$$C_2 = \frac{1}{2}\sum_{j=1}^{1} j^2\alpha_j - (\sum_{j=1}^{1} j\beta_j + \nu\beta_\nu) - (\sum_{j=0}^{1} \gamma_j + \gamma_\nu),$$

$$\vdots$$

$$C_q = \frac{1}{q!}[\sum_{j=1}^{1} j^q \alpha_j - q(\sum_{j=1}^{1} j^{q-1}\beta_j + \nu^{q-1}\beta_\nu) - q(q-1)(\sum_{j=1}^{1} j^{q-2}\gamma_j + \nu^{q-2}\gamma_\nu)].$$

According to Henrici [14], we say that the method (2) has order p if

$$C_0 = C_1 = \ldots = C_p = 0, \quad C_{p+1} \neq 0.$$

Therefore, C_{p+1} is the error constant and $C_{p+1}h^{p+1}y^{(p+1)}(x_n)$ the principal local truncation error at the point x_n. The local truncation error (LTE) is given by

$$LTE = C_{p+1}h^{p+1}y^{(p+1)}(x_n) + O(h^{p+2}).$$

It is established from our calculations that the IVMs (9) and (10) have order 6 and relatively small error constants as displayed in table 3.

Zero-stability. In order to analyze the methods for zero-stability, we write (9) to (10) as a block method given by the matrix difference equation

$$A^{(0)}Y_\mu = A^{(1)}Y_{\mu-1} + h[B^{(0)}F_\mu + B^{(1)}F_{\mu-1}] + h^2[C^{(0)}G_\mu + C^{(1)}G_{\mu-1}] \quad (12)$$

where $Y_\mu = (y_{n+\frac{1}{2}}, y_{n+1})^T$, $Y_{\mu-1} = (y_{n-\frac{1}{2}}, y_n)^T$

$$F_\mu = (f_{n+\frac{1}{2}}, f_{n+1})^T, \quad F_{\mu-1} = (f_{n-\frac{1}{2}}, f_n)^T,$$

$$G_\mu = (g_{n+\frac{1}{2}}, g_{n+1})^T, \quad G_{\mu-1} = (g_{n-\frac{1}{2}}, g_n)^T,$$

for $\mu = 1, \ldots$ and $n = 0, 1, \ldots$ and the matrices $A^{(0)}$, $A^{(1)}$, $B^{(0)}$, $B^{(1)}$, $C^{(0)}$, and $C^{(1)}$ are 2 by 2 matrices whose entries are given by the coefficients of (9) and (10). In particular, the matrices are defined as follows: $A^{(0)}$ is the identity matrix of dimension 2,

$$A^{(0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A^{(1)} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad B^{(0)} = \begin{pmatrix} 4/15 & 11/480 \\ 8/15 & 7/30 \end{pmatrix},$$

$$B^{(1)} = \begin{pmatrix} 0 & 101/480 \\ 0 & 7/30 \end{pmatrix}, \quad C^{(0)} = \begin{pmatrix} -1/24 & -1/320 \\ 0 & -1/60 \end{pmatrix},$$

$$C^{(1)} = \begin{pmatrix} 0 & 13/960 \\ 0 & 1/60 \end{pmatrix}.$$

It is worth noting that zero-stability is concerned with the stability of the difference system in the limit as h tends to zero. Thus, as $h \to 0$, the method (13) tends to the difference system

$$A^{(0)}Y_\mu - A^{(1)}Y_{\mu-1} = 0$$

whose first characteristic polynomial $\rho(R)$ is given by

$$\rho(R) = \det(RA^{(0)} - A^{(1)}) = R(R-1) \tag{13}$$

Following Fatunla[7], the block method (13) is zero-stable, since from (14), $\rho(R) = 0$ satisfy $|R_j| \leq 1$, $j = 1, 2$ and for those roots with $|R_j| = 1$, the multiplicity does not exceed 1. The block method (13) is consistent as it has order $p > 1$. According to Henrici[14], we can safely assert the convergence of the block method (13).

Linear-stability. A-stability is discussed in the spirit of [2], [13] where we consider the usual test equations

$$y' = \lambda y, \qquad y'' = \lambda^2 y$$

which are applied to (13) to yield

$$Y_\mu = M(q)Y_{\mu-1}\ ,\ q = \lambda h, \tag{14}$$

where the matrix $M(q)$ is given by

$$M(q) = (A^{(0)} - qB^{(0)} - q^2C^{(0)})^{-1}(A^{(1)} + qB^{(1)} + q^2C^{(1)}).$$

From (15) we obtain the usual property of A-stability, which requires that for all $q \in \mathbb{C}^-$, $M(q)$ must have a dominant eigenvalue λ_2 such that $|\lambda_2| < 1$. Our analysis show that the matrix M(q) has eigenvalues $\{\lambda_1, \lambda_2\} = \{0, \lambda_2\}$, where the dominant eigenvalue λ_2 is a function of q given by

$$\lambda_2 = \frac{P(q)}{P(-q)} \tag{15}$$

where $P(q) = 1 + \frac{1}{2}q + \frac{13}{120}q^2 + \frac{1}{80}q^3 + \frac{1}{1440}q^4$.

Hence, λ_2 is a [4/4] rational approximation of order six to e^q. It is obvious from (16) that for $Re(q) < 0$, $|\lambda_2| < 1$. The block method (13) is A-stable since its region of absolute stability contains the left half-plane $\mathbb{C}^-$. Therefore, there is no restriction on λh, which makes (13) a viable candidate for stiff problems.

Implementation. Our method is implemented more efficiently by combining methods (9) and (10) as simultaneous integrators for IVPs without requiring starting values and predictors. We proceed by explicitly obtaining initial conditions at $x_{n+1}, n = 0, 1, \ldots, N-1$ using the computed values $U(x_{n+1}) = y_{n+1}$ over sub-intervals $[x_0, x_1], \ldots, [x_{N-1}, x_N]$. For instance, $n = 0$, $\mu = 1$, $(y_{\frac{1}{2}}, y_1)^T$ are simultaneously obtained over the sub-interval $[x_0, x_1]$, as y_0 is known from the IVP, for $n = 1$, $\mu = 2$, $(y_{\frac{3}{2}}, y_2)^T$ are simultaneously obtained over the sub-interval $[x_1, x_2]$, as y_1 is known form the previous block, and so on. Hence, the sub-intervals do not over-lap and the solutions obtained in this manner are more accurate than those obtained in the conventional way. We note that for linear problems, we solve (1) directly with our Mathematica code enhanced by the feature *NSolve*[ ]. Nonlinear problems were solved by the Newton's method which uses the feature *FindRoot*[ ] (see Keiper and Gear [19]). It is vital to note that Mathematica can symbolically compute derivatives, hence the entries of the Jacobian matrix which involve the partial derivatives of both f and g are automatically generated.

4. Numerical Examples

In this section, we give numerical examples to illustrate the accuracy of the methods. We find absolute errors of the approximate solution on the partition π_N as $|y - y(x)|$. All computations were carried out using a written code in Mathematica 8.0.

Example 4.1. We consider the following IVP on the range $0 \leq x \leq 1$.

$$y' = -y + 95z, \;\; y(0) = 1$$

$$z' = -y - 97z, \;\; z(0) = 1$$

$$\text{Exact} : y(x) = \frac{95}{47}e^{-2x} - \frac{48}{47}e^{-96x} \;, \; z(x) = \frac{48}{47}e^{-96x} - \frac{1}{47}e^{-2x}$$

Jackson and Kenue [15] and Cash [4] also solved this problem using A-stable exponential fitting methods of the multiderivative type of orders 4 and 5. The HSDM is A-stable, has a sightly higher order of 6, and also of multiderivative type, hence it is comparable with the methods in [15] and [4]. The errors in the solution were obtained at $x = 1$ using the HSDM for fixed step-sizes $h = \{0.125, 0.0625, 0.03125\}$ as shown in Table 1. Similar results were obtained for the same problem by Cash [4] and Jackson and Kenue [15] as reproduced in Table 1. It is seen that the HSDM is more accurate than the methods in [4] and [15]. We remark that although the HSDM is expected to perform better because of its higher order, it is vital to note that even for a large step size of $h = 0.125$, the HSDM performs better than those in [4] and [15] with a smaller step size of $h = 0.0625$. Hence, for this example, the HSDM is superior pertaining to accuracy.

Step	Method	$y(1)$ ($\lvert error \rvert$)	$z(1) \times 10^2$ ($\lvert error \rvert$) $\times 10^2$
0.125	HSDM ($p = 6$)	0.27355004	-0.28794740
		(9×10^{-11})	(1×10^{-8})
0.0625	Jackson-Kenue ($p = 4$)	0.2735503	-0.2879477
		(3×10^{-7})	(4×10^{-7})
	Cash ($p = 4$)	0.2735498	-0.2879471
		(3×10^{-7})	(3×10^{-7})
	Cash ($p = 5$)	0.27355005	-0.28794742
		(1×10^{-8})	(1×10^{-8})
	HSDM ($p = 6$)	0.27355004	-0.28794741
		(3×10^{-12})	(3×10^{-12})
0.03125	Jackson-Kenue ($p = 4$)	0.27355005	-0.28794742
		(1×10^{-8})	(1×10^{-8})
	Cash ($p = 4$)	0.27355003	-0.28794740
		(1×10^{-8})	(1×10^{-8})
	HSDM ($p = 6$)	0.27355004	-0.28794741
		(5×10^{-14})	(5×10^{-14})
	True solution	0.27355004	$-0.28794741 \times 10^{-2}$

Table 1: A comparison of methods for Example 4.1

Example 4.2. Our second test example is the the following linear system which has been solved on the range $0 \leq x \leq 3$.

$$y_1' = -21y_1 + 19y_2 - 20y_3,\ y_1(0) = 1$$

$$y_2' = 19y_1 - 21y_2 + 20y_3,\ y_2 = 0$$

$$y_3' = 40y_1 - 40y_2 + 40y_3,\ y_3 = -1$$

The exact solution of the system is given by

$y_1(x) = \frac{1}{2}(e^{-2x} + e^{-40x}(\cos(40x) + \sin(40x)))$

$y_2(x) = \frac{1}{2}(e^{-2x} - e^{-40x}(\cos(40x) + \sin(40x)))$

$y_3(x) = \frac{1}{2}(2e^{-40x}(\sin(40x) - \cos(40x)))$

This problem has also been solved by Brugnano and Trigiante [2] using the Extended Trapezoidal Rules (ETRs), Extended Trapezoidal Rules of Second kind (ETR_2s), and Top Order Methods (TOMs). These methods have order 6 and hence comparable with the HSDM which is also of order 6. The results for the maximum absolute errors ($MaxErr = maximum|y(x) - y|$) are reproduced in table 2 and compared with the results given by our method. It is seen from

h	HSDM	ROC	ETRs	ROC	ETR_2s	ROC	TOMs	ROC
2×10^{-2}	9.335×10^{-7}		3.778×10^{-3}		3.513×10^{-3}		1.552×10^{-3}	
1×10^{-2}	1.401×10^{-8}	6.06	1.007×10^{-4}	5.23	8.615×10^{-5}	5.35	9.775×10^{-6}	7.31
5×10^{-3}	2.308×10^{-10}	5.92	1.091×10^{-6}	6.53	7.231×10^{-7}	6.90	1.197×10^{-7}	6.35
2.5×10^{-3}	3.598×10^{-12}	6.00	1.793×10^{-8}	5.93	8.864×10^{-9}	6.35	1.853×10^{-9}	6.01

Table 2: A comparison of methods using the maximum absolute errors ($MaxErr = maximum|y(x) - y|$) and ROC for Example 4.2

Method	HSDM (9)	HSDM (10)	ETRs	ETR_2s	TOMs
Error constant C_{p+1}	$-\frac{1}{7!120}$	$-\frac{1}{7!240}$	$-\frac{191}{7!12}$	$-\frac{6}{7!}$	$-\frac{9}{7!5}$

Table 3: The error constants of five methods of order six

table 2 that our method performs better than those in [2]. The accuracy of our method is further explained by the error constants compared in table 3. We also give the rate of convergence (ROC) which is calculated using the formula $ROC = \log_2(E^{2h}/E^h)$, E^h is the maximum absolute error obtained using the step size h. In all cases, the rate of convergence is consistent with the order of the methods. Thus, for this example, our method is superior in terms of accuracy.

Example 4.3. Our third test example is the given nonlinear IVP which was also solved by Norsett [24] and Jain[16] .

$$y' = -100xy^2,\ y(1) = 1/51,\ 0 \le x \le 20$$

$$\text{Exact} : y(x) = 1/(1 + 50x^2)$$

The HSDM performed excellently when tested on this nonlinear example. The methods given in [24] and [16] are A-stable and hence comparable with the HSDM which is also A-stable. The results produced by our methods were better than those given in [24] and [16]. We note that although our method is expected to perform better, because of its higher order, it is observed that even for a large step size of $h = \frac{1}{4}$, our method performs better than the methods in [24] and [16] with a smaller step size of $h = \frac{1}{16}$. Details of the numerical results are given in table 5.

h	x	HSDM ($p=6$) (y) (|$error$|)	Norsett (y)($p=6$) (|$error$|)	Jain (y)($p=4$) (|$error$|)	Exact ($y(x)$)
$\frac{1}{16}$	10	$0.19996001 \times 10^{-3}$ (6.163×10^{-15})	$0.19995554 \times 10^{-3}$ (4.470×10^{-9})	$0.19996018 \times 10^{-3}$ (1.700×10^{-10})	$0.19996001 \times 10^{-3}$
$\frac{1}{8}$	10	$0.19996001 \times 10^{-3}$ (5.735×10^{-14})	$0.19991486 \times 10^{-3}$ (4.515×10^{-8})	$0.19996310 \times 10^{-3}$ (3.090×10^{-9})	$0.19996001 \times 10^{-3}$
	20	$0.49997500 \times 10^{-4}$ (1.853×10^{-14})	$0.49994562 \times 10^{-4}$ (2.938×10^{-9})	$0.49997695 \times 10^{-4}$ (1.950×10^{-10})	$0.49997500 \times 10^{-4}$
$\frac{1}{4}$	10	$0.19996001 \times 10^{-3}$ (3.664×10^{-12})	$0.19906134 \times 10^{-3}$ (8.987×10^{-7})	$0.20000938 \times 10^{-3}$ (4.937×10^{-8})	$0.19996001 \times 10^{-3}$
	20	$0.49997500 \times 10^{-4}$ (3.238×10^{-13})	$0.49940176 \times 10^{-4}$ (5.732×10^{-8})	$0.50000607 \times 10^{-4}$ (3.107×10^{-9})	$0.49997500 \times 10^{-4}$

Table 4: Exact solution $y(x)$, approximate solution y, and absolute errors, $|y(x) - y|$ for three methods for Example 4.3

Example 4.4. As our fourth test example, we consider the following IVP considered by Wu and Xia [31].

$$y_1' = -1002y_1 + 1000y_2^2,\ y_1(0) = 1$$

$$y_2' = y_1 - y_2(1 + y_2),\ y_2(0)) = 1$$

$$\text{Exact}: y_1(x) = e^{-2x}\ ,\ y_2(x) = e^{-x}$$

The method given in Wu and Xia [31] is a low accuracy method which was shown to perform efficiently on stiff systems, hence we chose it for comparison with the HSDM. It is obvious from the numerical results in table 5 that HSDM performed excellently for step sizes $h = \{0.1, 0.01\}$ compared with the method in Wu and Xia [31] where step sizes $h = \{0.002, 0.001\}$ were used. Details of the numerical results are given in table 5.

Example 4.5. We consider the given non-linear system on the range $0 \le x \le 48$.

$$y_1' = -0.013y_2 - 1000y_1y_2 - 2500y_1y_3,\ y_1(0) = 0$$

$$y_2' = -0.013y_2 - 1000y_1y_2,\ y_2(0) = 1$$

	Wu and Xia [31]				Our Method			
x	h	N	$Err(y_1)$	$Err(y_2)$	h	N	$Err(y_1)$	$Err(y_2)$
1	0.002	500	2.5606×10^{-7}	8.0150×10^{-8}	0.1	10	5.6763×10^{-13}	6.5675×10^{-13}
10	0.001	10000	5.5468×10^{-16}	6.0936×10^{-12}	0.01	1000	7.0972×10^{-22}	7.8198×10^{-18}

Table 5: Absolute Errors, ($Err(y_1) = |y(x)-y_1|$, $Err(y_2) = |y(x)-y_2|$), for Example 4.4

$$y_3' = -2500y_1y_3,\ y_3(0) = 1$$

h	x	$Err(y_1)$	$Err(y_2)$	$Err(y_3)$
1/8	2	9.850×10^{-7}	4.939×10^{-5}	4.840×10^{-5}
	48	1.918×10^{-10}	4.920×10^{-5}	4.920×10^{-5}
1/16	2	1.927×10^{-8}	4.198×10^{-6}	4.179×10^{-6}
	48	1.205×10^{-11}	3.092×10^{-6}	3.092×10^{-6}
1/32	2	1.370×10^{-12}	2.629×10^{-7}	2.629×10^{-7}
	48	7.517×10^{-13}	1.928×10^{-7}	1.928×10^{-7}
1/64	2	8.465×10^{-14}	1.621×10^{-8}	1.621×10^{-8}
	48	4.634×10^{-14}	1.189×10^{-8}	1.189×10^{-8}

Table 6: Absolute Errors, ($Err(y_i) = |y(x) - y_i|$), $i = 1, 2, 3$, for Example 4.5

True solution:

$$y_i(2) = \{-3.616933169289 \times 10^{-6}, 0.9815029948230, 1.018493388244\},$$

$$y_i(48) = \{-1.945338956808 \times 10^{-6}, 0.6110474831446, 1.388950571516\},\quad i = 1, 2, 3$$

(see Jeltsch[18]).

The accuracy of the HSDM for this example was measured by computing the absolute errors at $x = \{2, 48\}$ and the detailed results are displayed in table 6.

Example 4.6. We consider the given linear stiff system on the range $0 \leq x \leq 10$ (see Enright [6])

$$y_1' = -0.1y_1,\ y_1(0) = 1, \quad y_2' = -10y_2,\ y_2(0) = 1$$

$$y_3' = -100y_3,\ y_3(0) = 1, \quad y_4' = -1000y_4,\ y_4(0) = 1$$

Method	Steps	Function Calls	Jacobian Calls	MaxErr
Enright	125	384	384	0.44×10^{-6}
Gear	422	1111	18	1.06×10^{-6}
HSDM	10	88	10	6.09×10^{-13}

Table 7: Errors (MaxErr) for Example 4.6

The accuracy of the HSDM for this example was measured by computing the maximum absolute errors on the given interval. The efficiency of the method was measured using the number of function evaluations involved in the computations. In table 7, it is seen that the maximum absolute errors and the number of function evaluations used to generate the errors for different number of steps are small, hence the method is accurate and efficient. It is also observed that the HSDM performs better than the methods given in [6].

Example 4.7. We consider the given linear stiff system on the range $0 \leq x \leq 20$ which is highly oscillatory with all of its eigenvalues near the imaginary axis.

$$y_1' = -10y_1 + 50y_2,\ y_1(0) = 0, \quad y_2' = -50y_1 - 10y_2,\ y_2(0) = 1$$

$$y_3' = -40y_3 + 200y_4,\ y_3(0) = 0, \quad y_4' = -200y_3 - 40y_4,\ y_4(0) = 1$$

$$y_5' = -0.2y_5 - 2y_6,\ y_5(0) = 0, \quad y_6' = -2y_5 - 0.2y_6,\ y_6(0) = 1$$

This problem has also been solved by Rockswold [27] using four software packages as follows:

- The GEAR package which uses the Backward Differentiation Formulas (BDFs) of orders 1-5.
- The ISU package which uses the BDFs of orders 1-4.

- The αTF package which uses a set of α-multistep methods of orders 1-4.
- The VARBDF package which which uses the BDFs of orders 1-4.

The results given in [27] are reproduced in table 8 and compared with the results given by the HSDM, using the number of steps required and the maximum global error. It is seen from table 8 that the HSDM performs better than the GEAR, ISU, and VARBDF packages. It is also observed that the HSDM is highly competitive with the αTF package. These methods were chosen for comparison with the HSDM, because they are very popular for solving stiff systems. We note that the four packages are based on a variable order variable-step implementation, while the HSDM is a fixed step fixed order method. Hence, for this example the HSDM is very competitive.

5. Conclusion

A continuous representation of a HSDM with two 'off-step' points is proposed and used to obtained discrete methods (IVMs) which are combined and implemented as a block method for solving IVPs. The IVMs are assembled into a single block matrix equation which is A-stable, hence the block method is an excellent candidate for solving stiff problems. In particular, the method is implemented without the need for starting values or predictors, therefore complicated subroutines are avoided. We have demonstrated the accuracy of the method on seven problems (see tables 1, 2, 4, 5, 6, 7, 8). The numerical results show that our method is efficient and highly competitive with the existing methods cited in this paper. Our future research will be focused on developing and implementing variable order, variable step methods with global error estimates.

References

[1] K.E. Atkinson, *An Introduction to Numerical Analysis*, Second Edition, John Wiley and Sons, New York (1989).

[2] L. Brugnano, D. Trigiante, *Solving Differential Problems by Multitep Initial and Boundary Value Methods*, Gordon and Breach Science Publishers, Amsterdam (1998).

Method	Tolerance	Steps	MaxErr
GEAR	10^{-6}	1184	1.6×10^{-5}
	10^{-4}	824	1.5×10^{-3}
	10^{-2}	150	1.1×10^{-1}
ISU	10^{-6}	1176	7.7×10^{-5}
	10^{-4}	3644	3.2×10^{-3}
	10^{-2}	174	7.2×10^{-2}
VARBDF	10^{-6}	1176	2.9×10^{-5}
	10^{-4}	983	1.0×10^{-3}
	10^{-2}	903	2.3×10^{-1}
αTF	10^{-6}	826	3.1×10^{-5}
	10^{-4}	308	1.8×10^{-3}
	10^{-2}	127	6.3×10^{-2}
HSDM	unspecified	1000	3.3×10^{-6}
	unspecified	800	1.2×10^{-5}
	unspecified	350	9.6×10^{-4}
	unspecified	150	7.4×10^{-2}

Table 8: A comparison of methods ($MaxErr = maximum|y(x) - y|$) for Example 4.7

[3] J.C. Butcher, A modified multistep method for the numerical integration of ordinary differential equations, *J. Assoc. Comput. Mach.*, **12** (1965), 124-135.

[4] J.R. Cash, On the exponential fitting of composite multiderivative linear multistep methods, *SIAM J. Numer. Anal.*, **18** (1981), 808-821.

[5] G.G. Dahlquist, Numerical integration of ordinary differential equations, *Math. Scand.*, **4** (1956), 69-86.

[6] W.H. Enright, Second derivative multistep methods for stiff ordinary differential equations, *SIAM J. Numer. Anal.*, **11** (1974), 321-331.

[7] S.O. Fatunla, Block methods for second order IVPs, *Intern. J. Comput. Math.*, **41** (1991). 55-63.

[8] S.O. Fatunla, Numerical integrators for stiff and highly oscillatory differential equations, *Math. Comp.*, **34** (1980), 373-390.

[9] C.W. Gear, Hybrid methods for initial value problems in ordinary differential equations, *SIAM J. Numer. Anal.*, **2** (1965), 69-86.

[10] I. Gladwell, D.K. Sayers, Eds. *Computational Techniques for Ordinary Differential Equations*, Academic Press, New York (1976).

[11] W. Gragg, H.J. Stetter, Generalized multistep predictor-corrector methods, *J. Assoc. Comput. Mach.*, **11** (1964), 188-209.

[12] G.K. Gupta, Implementing second-derivative multistep methods using Nordsieck polynomial representation, *Math. Comp.*, **32** (1978), 13-18.

[13] E. Hairer, G. Wanner, *Solving Ordinary Differential Equations II*, Springer, New York (1996).

[14] P. Henrici, *Discrete Variable Methods in ODEs*, John Wiley, New York (1962).

[15] L.W. Jackson, S.K. Kenue, A fourth order exponentially fitted method, *SIAM J. Numer. Anal.*, **11** (1974), 965-978.

[16] R.K. Jain, Some *A*-stable methods for stiff ordinary differential equations, *Math. Comp.*, **26** (1972), 71-77.

[17] S.N. Jator, On the hybrid method with three-off step points for initial value problems, *International Journal of Mathematical Education in Science and Technology*, **41** (2010), 110-118.

[18] R. Jeltsch, Multistep methods using higher derivatives and damping at infinity, *Math. Comp.*, **31** (1977), 124-138.

[19] J.B. Keiper, C.W. Gear, The analysis of generalized backwards difference formula methods applied to Hessenberg form differential-algebraic equations, *SIAM J. Numer. Anal.*, **28** (1991), 833-858.

[20] J.J. Kohfeld, G.T. Thompson, Multistep methods with modified predictors and correctors, *J. Assoc. Comput. Mach.*, **14** (1967), 155-166.

[21] J.D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley, New York (1991).

[22] J.D. Lambert, *Computational Methods in Ordinary Differential Equations*, John Wiley, New York (1973).

[23] I. Lie, S.P. Norsett, Superconvergence for Multistep Collocation, *Math. Comp.*, **52** (1989), 65-79.

[24] S.P. Norsett, An A-stable modification of the Adams-Bashforth methods, In: *Conference on the Numerical Solution of Differential Equations*, Dundee, Scotland (1969).

[25] W.E. Milne, *Numerical Solution of Differential Equations*, John Wiley and Sons (1953).

[26] P. Onumanyi, D.O. Awoyemi, S.N. Jator, U.W. Sirisena, New linear mutlistep methods with continuous coefficients for first order initial value problems, *J. Nig. Math. Soc.*, **13** (1994), 37-51.

[27] G.K. Rockswold, Implementation of α-type multistep methods for stiff diffrential equations, *J. Comput. Appl. Math.*, **22** (1988), 63-69.

[28] J.D. Rosser, A Runge-kutta for all seasons, *SIAM, Rev.*, **9** (1967), 417-452.

[29] D. Sarafyan, Multistep methods for the numerical solution of ordinary differential equations made self-starting, *Tech. Report*, **495**, Math. Res. Center, Madison (1965).

[30] L.F. Shampine, H.A. Watts, Block implicit one-step methods, *Math. Comp.*, **23** (1969), 731-740.

[31] X.Y. Wu, J.L. Xia, Two low accuracy methods for stiff systems, *Appl. Math. Comp.*, **123** (2001), 141-153.