Numer Algor (2013) 62:13–26
DOI 10.1007/s11075-012-9562-1

ORIGINAL PAPER

Trigonometrically fitted block Numerov type method for $y'' = f(x, y, y')$

Samuel N. Jator · S. Swindell · R. French

Received: 30 January 2012 / Accepted: 12 March 2012 /
Published online: 24 March 2012

Abstract A trigonometrically fitted block Numerov type method (TBNM), is proposed for solving $y'' = f(x, y, y')$ directly without reducing it to an equivalent first order system. This is achieved by constructing a continuous representation of the trigonometrically fitted Numerov method (CTNM) and using it to generate the well known trigonometrically fitted Numerov method (TNUM) and three new additional methods, which are combined and applied in block form as simultaneous numerical integrators. The stability property of the TBNM is discussed and the performance of the method is demonstrated on some numerical examples to show accuracy and efficiency advantages.

Keywords Second order · Initial value problems · Trigonometrically fitted method · Block form

1 Introduction

The second order initial value problem (IVP) of the form

$$y'' = f(x, y), \ y(a) = y_0, \ y'(a) = y'_0, \tag{1}$$

The first author was supported by Austin Peay State University, Clarksville, TN, USA through the Faculty Development Leave, Spring 2010.

S. N. Jator (✉) · R. French
Department of Mathematics and Statistics, Austin Peay State University, Clarksville, TN 37044, USA
e-mail: Jators@apsu.edu

S. Swindell
Department of Physics and Astronomy, Austin Peay State University, Clarksville, TN 37044, USA

in which the first derivative does not appear explicitly is encountered in several areas of engineering and science, such as celestial mechanics, circuit theory, control theory, chemical kinetics, and biology. Several techniques for the direct solution of (1) have been investigated including linear multistep methods (LMMs) [17, 27, 28], multistep collocation methods (see [1, 5]), exponentially-fitting and trigonometrically-fitted methods [6, 8, 9, 12, 21, 26, 28], Runge-Kutta-Nyström methods (RKN) (see [10, 11, 25]).

Despite the numerous methods available for solving (1) directly, there are fewer methods available for directly solving the general second order IVPs of the form

$$y'' = f(x, y, y'),\ \ y(a) = y_0,\ \ y'(a) = y'_0, \tag{2}$$

in which the first derivative appears explicitly. In practice, (2) is solved by first reducing it to an equivalent first order system and solved by the numerous methods available for solving first order IVPs. Some direct methods available for (2) are due to [2, 9, 29, 30]. Most of these methods are implemented in a step-by-step fashion in which on the partition π_N, an approximation is obtained at x_n only after an approximation at x_{n-1} has been computed, where

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

$h = \frac{b-a}{N}$ is the constant step-size of the partition π_N, N is a positive integer, and n is the grid index.

Recently, Jator [13, 15] and Jator and Li [14], solved (2) directly via methods of the linear multistep type. In this paper, a TBNM is proposed for solving (2) in which (1) is a special case, directly without reducing it to an equivalent first order system. This is achieved by constructing a continuous representation of the CTNM and using it to generate the well known TNUM and three new additional methods, which are combined and applied in block form as simultaneous numerical integrators for (2). We note that the concept of combining the main and additional methods for first order systems of differential equations is extensively discussed in [3]. We emphasize that the TBNM is applied as a block method to simultaneously produce approximations

$$\{y_{n+1}, y_{n+2}\} \text{ and } \left\{y'_{n+1}, y'_{n+2}\right\} \text{ at the points } \{x_{n+1}, x_{n+2}\} \text{ to the exact solutions}$$
$$\{y(x_{n+1}), y(x_{n+2})\} \text{ and } \{y'(x_{n+1}), y'(x_{n+2})\}, n = 0, 2, \ldots, N-2.$$

We note that in order to apply the block method at the next block to obtain $\{y_{n+3}, y_{n+4}\}$, the only necessary starting value is y_{n+2}, and the loss of accuracy in y_{n+2}, 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 since at all blocks except the first, the first function evaluation is already available from the previous block. Block methods are due to [19, 23, 24]. It is crucial to observe that the TBNM preserves the Runge-kutta traditional advantage of being self-starting and is more efficient, since it requires only one function evaluation per integration step.

The paper is organized as follows. In Section 2 we derive an approximation $U(x)$ for $y(x)$, given by CTNM which is used to obtain the TNUM and three new additional methods given in Section 3. The stability property and the computational aspect of the TBNM are given in Section 4. Numerical examples are given in Section 5 to show the accuracy and efficiency advantages. Finally, the conclusion of the paper is discussed in Section 6.

2 CTNM

In this section, we develop a CTNM on the interval from x_n to $x_{n+2} = x_n + 2h$, where h is the chosen step-length. In particular, we assume that the exact solution $y(x)$ on the interval $[x_n, x_{n+2}]$ is locally represented by $U(x)$ given by

$$U(x) = \alpha_0(x) y_n + \alpha_1(x) y_{n+1} + h^2(\beta_0(x) f_n + \beta_1(x) f_{n+1} + \beta_2(x) f_{n+2}), \tag{3}$$

where $\alpha_0(x),\ \alpha_1(x),\ \beta_j(x),\ j = 0, 1, 2$ are continuous coefficients that must be uniquely determined. We assume that $y_{n+j} = U(x_n + jh)$ is the numerical approximation to the analytical solution $y(x_{n+j})$, $y'_{n+j} = U'(x_n + jh)$ is an approximation to $y'(x_{n+j})$, $f_{n+j} = U''(x_n + jh)$ is an approximation to $y''(x_{n+j})$. We note that $f_{n+j} = f(x_{n+j}, y_{n+j}, y'_{n+j})$, $j = 0, 1, 2$.

Since the function (3) must pass through the points (x_n, y_n), (x_{n+1}, y_{n+1}), (x_{n+2}, y_{n+2}), we demand that the following five equations must be satisfied.

$$U\left(x_{n+j}\right) = y_{n+j},\ j = 0, 1, \qquad U''\left(x_{n+j}\right) = f_{n+j}, \quad j = 0, 1, 2. \tag{4}$$

Equation (4) leads to a system of five equations and five unknown parameters to be determined. In order to solve this system, we require that the method (3) be defined by the assumed basis functions

$$\alpha_j(x) = \sum_{i=0}^{4} \alpha_{i+1,j} P_i(x),\ j\epsilon\{0, 1\}; \quad h^2\beta_j(x) = \sum_{i=0}^{4} h^2\beta_{i+1,j} P_i(x),\ j\epsilon\{0, 1, 2\}, \tag{5}$$

where $P_i = \{1, x, x^2, \sin wx, \cos wx\}$ and the constants $\alpha_{i+1,j}$ and $h^2\beta_{i+1,j}$ are undetermined elements of the 5×5 matrix E, given by

$$E = \begin{pmatrix} \alpha_{1,0} & \alpha_{1,1} & h^2\beta_{1,0} & h^2\beta_{1,1} & h^2\beta_{1,2} \\ \alpha_{2,0} & \alpha_{2,1} & h^2\beta_{2,0} & h^2\beta_{2,1} & h^2\beta_{2,2} \\ \alpha_{3,0} & \alpha_{3,1} & h^2\beta_{3,0} & h^2\beta_{3,1} & h^2\beta_{3,2} \\ \alpha_{4,0} & \alpha_{4,1} & h^2\beta_{4,0} & h^2\beta_{4,1} & h^2\beta_{4,2} \\ \alpha_{5,0} & \alpha_{5,1} & h^2\beta_{5,0} & h^2\beta_{5,1} & h^2\beta_{5,2}. \end{pmatrix}$$

We also define the matrix W as

$$W = \begin{pmatrix} P_0(x_n) & \cdots & P_4(x_n) \\ P_0(x_{n+1}) & \cdots & P_4(x_{n+1}) \\ P_0''(x_n) & \cdots & P_4''(x_n) \\ P_0''(x_{n+1}) & \cdots & P_4''(x_{n+1}) \\ P_0''(x_{n+2}) & \cdots & P_4''(x_{n+2}). \end{pmatrix}$$

We consider further notations by defining the following vectors:

$$V = (y_n, y_{n+1}, f_n, f_{n+1}, f_{n+2})^T, \quad P(x) = (P_0(x), P_1(x), P_2(x), P_3(x), P_4(x))^T,$$

where T denotes the transpose of the vectors.

Theorem 2.1 *Let $U(x)$ satisfy conditions (4) and let W be invertible, then the method (3) is equivalent to*

$$U(x) = V^T \left(W^{-1}\right)^T P(x).$$

Proof The proof takes the form given in [20] with obvious notational modifications. We begin by substituting (5) into (3) to yield

$$U(x) = \sum_{i=0}^{4} \Theta_i P_i(x), \tag{6}$$

where $\Theta_i = \sum_{j=0}^{1} \alpha_{i+1,j} y_{n+j} + \sum_{j=0}^{2} h^2 \beta_{i+1,j} f_{n+j}$ are undetermined coefficients that can be written in vector form as

$$\Theta = (\Theta_0, \Theta_1, \Theta_2, \Theta_3, \Theta_4)^T.$$

Furthermore, we demand that (6) satisfies (4) to obtain the system

$$W\Theta = V,$$

which produces

$$\Theta = W^{-1} V. \tag{7}$$

We note that Θ can easily be expressed in terms of E as $\Theta = EV \Rightarrow E = W^{-1}$. Next, we write (6) in vector form as

$$U(x) = \Theta^T P(x). \tag{8}$$

Then substituting (7) into (8) and simplifying we obtain

$$U(x) = V^T \left(W^{-1}\right)^T P(x). \tag{9}$$

The proof is complete. □

3 TNUM and additional methods

TNUM The well known TNUM is obtained by evaluating (9) at $x = x_{n+2}$. Thus, $y_{n+2} = U(x_n + 2h)$ gives the following method.

$$y_{n+2} - 2y_{n+1} + y_n = h^2(\beta_0 f_n + \beta_1 f_{n+1} + \beta_2 f_{n+2}), \tag{10}$$

where

$$\begin{cases} \beta_0 = \left(-4\csc^2(u/2) + 2u^2\csc^2(u/2) + 4\cos u \csc^2(u/2)\right) / \left(8u^2\right), \\ \beta_1 = \left(4\csc^2(u/2) - 4\cos u\csc^2(u/2) - 2u^2\cos u\csc^2(u/2)\right) / \left(4u^2\right), \\ \beta_2 = \left(-2\csc^2(u/2) + 2u^2\csc^2(u/2) + 2\cos(3u/2)\csc^2(u/2)\sec(u/2)\right) / \left(8u^2\right), \end{cases} \tag{11}$$

and $u = wh$.

The Taylor series is used for small values of u (see [26]). Thus, the coefficients can be expressed as

$$\begin{cases} \beta_0 = \frac{1}{12} + \frac{u^2}{240} + \frac{u^4}{6048} + \frac{u^6}{172800} + \frac{u^8}{5322240}, \\ \beta_1 = \frac{5}{6} - \frac{u^2}{120} - \frac{u^4}{3024} - \frac{u^6}{86400} - \frac{u^8}{2661120}, \\ \beta_2 = \frac{1}{12} + \frac{u^2}{240} + \frac{u^4}{6048} + \frac{u^6}{172800} + \frac{u^8}{5322240}. \end{cases} \tag{12}$$

It is vital to note that when $u = 0$, (10) is the standard Numerov method. The discretization of (1) using (10) gives more unknowns than equations which if solved will lead to an indeterminate. Hence, we are compelled to look for additional methods. Fortunately, (3) is continuous and is used to provide the needed methods via its first derivative given by (13).

$$U'(x) = \frac{d}{dx}\left(\alpha_0(x)y_n + \alpha_1(x)y_{n+1} + h^2\sum_{j=0}^{2}\beta_j(x)f_{n+j}\right) \tag{13}$$

Additional methods The additional methods (14), (17), and (20) are obtained from (13) by demanding that $y'_n = U'(x_n)$, $y'_{n+1} = U'(x_n + h)$, and $y'_{n+2} = U'(x_n + 2h)$.

Evaluating (13) at $x = x_n$ and noting that $y'_n = U'(x_n)$, we obtain

$$hy'_n = y_{n+1} - y_n + h^2\left(\beta_{0,0}f_n + \beta_{1,0}f_{n+1} + \beta_{2,0}f_{n+2}\right), \tag{14}$$

where

$$\begin{cases} \beta_{0,0} = \left(-4\csc^2(u/2) - u^2\csc^2(u/2) + 4\cos u\csc^2(u/2) + 2u\cot u\csc^2(u/2)\right. \\ \qquad\left. - 2u\cos(2u)\csc^2(u/2)\csc u\right) / \left(8u^2\right), \\ \beta_{1,0} = \left(2\csc^2(u/2) - 2\cos u\csc^2(u/2) + u^2\cos u\csc^2(u/2)\right. \\ \qquad\left. -2u\csc^2(u/2)\sin u\right) / \left(4u^2\right), \\ \beta_{2,0} = \left(-u^2\csc^2(u/2) + 2u\csc(u/2)\sec(u/2)\right) / \left(8u^2\right). \end{cases} \tag{15}$$

The Taylor series is used for small values of u. Thus, the coefficients can be expressed as

$$\begin{cases} \beta_{0,0} = -\dfrac{7}{24} - \dfrac{7u^2}{480} - \dfrac{71u^4}{60480} - \dfrac{53u^6}{483840} - \dfrac{23u^8}{2128896}, \\ \beta_{1,0} = -\dfrac{1}{4} + \dfrac{u^2}{144} + \dfrac{u^4}{4320} + \dfrac{u^6}{134400} + \dfrac{u^8}{4354560}, \\ \beta_{2,0} = \dfrac{1}{24} + \dfrac{11u^2}{1440} + \dfrac{19u^4}{20160} + \dfrac{247u^6}{2419200} + \dfrac{1013u^8}{95800320}. \end{cases} \tag{16}$$

Evaluating (13) at $x = x_{n+1}$ and noting that $y'_{n+1} = U'(x_n + h)$, we obtain

$$hy'_{n+1} = y_{n+1} - y_n + h^2(\beta_{0,1} f_n + \beta_{1,1} f_{n+1} + \beta_{2,1} f_{n+2}), \tag{17}$$

where

$$\begin{cases} \beta_{0,1} = \left(-4\csc^2(u/2) + u^2\csc^2(u/2) + 4\cos u\csc^2(u/2) - 2u\cot u\csc^2(u/2) \right. \\ \qquad \left. +2u\csc^2(u/2)\csc u\right) / \left(8u^2\right), \\ \beta_{1,1} = \left(2\csc^2(u/2) - 2\cos u\csc^2(u/2) - u^2\cos u\csc^2(u/2)\right) / \left(4u^2\right), \\ \beta_{2,1} = \left(u^2\csc^2(u/2) - 2u\csc(u/2)\sec(u/2)\right) / \left(8u^2\right). \end{cases} \tag{18}$$

The Taylor series is used for small values of u. Thus, the coefficients can be expressed as

$$\begin{cases} \beta_{0,1} = \dfrac{1}{8} + \dfrac{17u^2}{1440} + \dfrac{67u^4}{60480} + \dfrac{29u^6}{268800} + \dfrac{1031u^8}{95800320}, \\ \beta_{1,1} = \dfrac{5}{12} - \dfrac{u^2}{240} - \dfrac{u^4}{6048} - \dfrac{u^6}{172800} - \dfrac{u^8}{5322240}, \\ \beta_{2,1} = -\dfrac{1}{24} - \dfrac{11u^2}{1440} - \dfrac{19u^4}{20160} - \dfrac{247u^6}{2419200} - \dfrac{1013u^8}{95800320}. \end{cases} \tag{19}$$

Evaluating (13) at $x = x_{n+2}$ and noting that $y'_{n+2} = U'(x_n + 2h)$, we obtain

$$hy'_{n+2} = y_{n+1} - y_n + h^2(\beta_{0,2} f_n + \beta_{1,2} f_{n+1} + \beta_{2,2} f_{n+2}) \tag{20}$$

where

$$\begin{cases} \beta_{0,2} = \left(-4\csc^2(u/2) + 3u^2\csc^2(u/2) + 4\cos u\csc^2(u/2) + 2u\cot u\csc^2(u/2) \right. \\ \qquad \left. - 2u\csc^2(u/2)\csc u\right) / \left(8u^2\right), \\ \beta_{1,2} = \left(2\csc^2(u/2) - 2\cos u\csc^2(u/2) - 3u^2\cos u\csc^2(u/2) \right. \\ \qquad \left. + 2u\csc^2(u/2)\sin u\right) / \left(4u^2\right), \\ \beta_{2,2} = \left(3u^2\csc^2(u/2) - 2u\csc^2(u/2)\sec(u/2)\sin(3u/2)\right) / \left(8u^2\right). \end{cases} \tag{21}$$

The Taylor series is used for small values of u. Thus, the coefficients can be expressed as

$$\begin{cases} \beta_{0,2} = \frac{1}{24} - \frac{u^2}{288} - \frac{47u^4}{60480} - \frac{233u^6}{2419200} - \frac{199u^8}{19160064}, \\ \beta_{1,2} = \frac{13}{12} - \frac{11u^2}{720} - \frac{17u^4}{30240} - \frac{23u^6}{1209600} - \frac{29u^8}{47900160}, \\ \beta_{2,2} = \frac{3}{8} + \frac{3u^2}{160} + \frac{3u^4}{2240} + \frac{31u^6}{268800} + \frac{13u^8}{1182720}. \end{cases} \tag{22}$$

Local Truncation Error The Local Truncation Errors (LTEs) for methods (10), (14), (17), and (20); denoted as LTE(10), LTE(14), LTE(17), and LTE(20) are given by

$$\begin{cases} LTE(10) = -\frac{h^6}{240}\left(w^2 y^{(4)}(x_n) + y^{(6)}(x_n)\right), \\ LTE(14) = -\frac{3h^6}{160}\left(w^2 y^{(4)}(x_n) + y^{(6)}(x_n)\right), \\ LTE(17) = \frac{5h^6}{288}\left(w^2 y^{(4)}(x_n) + y^{(6)}(x_n)\right), \\ LTE(20) = -\frac{43h^6}{1440}\left(w^2 y^{(4)}(x_n) + y^{(6)}(x_n)\right). \end{cases} \tag{23}$$

4 TBNM

In this section the methods (10), (14), (17), and (20) are combined to give the TBNM, which takes form of a general linear method of Butcher [4]. We then define the block-by-block method as a method for computing vectors $Y_0, Y_1, \ldots$ in sequence (see [7]). Let the ν-vectors ($\nu = 2$ is the number of points within the block) Y_μ, $Y_{\mu-1}$, F_μ, and $F_{\mu-1}$ for $\mu = m\nu$, $m = 0, 1, \ldots$ be given as $Y_\mu = (y_{n+1}, y_{n+2}, hy'_{n+1}, hy'_{n+2})^T$, $Y_{\mu-1} = (y_{n-1}, y_n, hy'_{n-1}, hy'_n)^T$, $F_\mu = (f_{n+1}, f_{n+2}, hf'_{n+1}, hf'_{n+2})^T$, and $F_{\mu-1} = (f_{n-1}, f_n, hf'_{n-1}, hf'_n)^T$, then the 1-block 2-point method for (1) and (2) is given by

$$Y_\mu = \sum_{i=1}^{1} A^{(i)} Y_{\mu-i} + h^2 \sum_{i=0}^{1} B^{(i)} F_{\mu-i}, \tag{24}$$

where $A^{(i)}$, $B^{(i)}$, $i = 0, 1$ are 4 by 4 matrices whose entries are given by the coefficients of (10), (14), (17), and (20).

Definition 4.1 The block method (24) is zero stable provided the roots R_j, $j = 1, \ldots, 4$ of the first characteristic polynomial $\rho(R)$ specified by

$$\rho(R) = \det\left[\sum_{i=0}^{1} A^{(i)} R^{1-i}\right] = 0,\ A^{(0)} = -I \tag{25}$$

satisfies $|R_j| \leq 1$, $j = 1, \ldots, 4$ and for those roots with $|R_j| = 1$, the multiplicity does not exceed 2 (see [7]).

Consistency of TBNM We note that the block method (24) is consistent as it has order $p > 1$. It is easily seen from (25) and invoking Definition 4.1 that the block method (24) is zero-stable since for $\rho(R) = R^2(R-1)^2$, $\rho(R) = 0$ satisfies $|R_j| \leq 1$, $j = 1, \ldots, 4$, and for those roots with $|R_j| = 1$, the multiplicity does not exceed 2. Hence, the block method (24) is convergent since consistency + zero-stability = convergence.

Linear stability of the TBNM The linear-stability of the TBNM is discussed by applying the method to the test equation $y'' = \lambda y$, where λ is expected to run through the (negative) eigenvalues of the Jacobian matrix $\frac{\partial f}{\partial y}$ (see [25]). Letting $q = \lambda h^2$ and $u = wh$, it is easily shown that the application of (24) to the test equation yields

$$Y_\mu = M(q; u) Y_{\mu-1} \, , \, M(q; u) := \left(A^{(0)} - qB^{(0)}\right)^{-1} \left(A^{(1)} + qB^{(1)}\right), \tag{26}$$

where the matrix $M(q; u)$ is the amplification matrix which determines the stability of the method.

Definition 4.2 A region of stability is a region in the $q - u$ plane, throughout which $|\rho(q; u)| \leq 1$, where $\rho(q; u)$ is the spectral radius of $M(q; u)$ (see [6]).

We observed that in the $q - u$ plane the TBNM is stable for $q \in [0, 12]$ and $u \in [-\pi, \pi]$ (see Example 5.5).

Computational aspects We use the main method (10) and the additional methods (14), (17), and (20) to simultaneously obtain the approximations $(y_{n+1}, y_{n+2})^T$ and $(y'_{n+1}, y'_{n+2})^T$, $n = 0, 2, \ldots, N-2$ over sub-intervals $[x_0, x_2], \ldots, [x_{N-2}, x_N]$. For instance, $n = 0$, $(y_1, y_2)^T$ and $(y'_1, y'_2)^T$ are simultaneously obtained over the sub-interval $[x_0, x_2]$, as y_0 and y'_0 are known from the IVP (2), for $n = 2$, $(y_3, y_4)^T$ and $(y'_3, y'_4)^T$ are simultaneously obtained over the sub-interval $[x_2, x_4]$, as y_2 and y'_2 are known from the previous block, and so on.

The computations were carried out using a code written in Mathematica 8.0. We note that for linear problems, the code was enhanced by the feature *NSolve*[ ] and for nonlinear problems, the Newton's method was used enhanced by *FindRoot*[ ] (see [16]).

5 Numerical Examples

In this section, we give some numerical examples to illustrate the accuracy (small errors) and efficiency (fewer number of function evaluations (FNCS)) of the TBNM. We find the absolute error of the approximate solution on the partition π_N as $|y - y(x)|$. We note that the method requires only two function evaluations per block except in the first block where three function evaluations are used. Thus, in general, the method requires a total of $(N + 1)$ FNCS. All computations were carried out using our written code in Mathematica 8.0.

5.1 Problems where y' does not appear explicitly.

In this subsection, the TBNM is compared with some existing methods in the literature.

Example 5.1 We consider the given inhomogeneous IVP (see [26])

$$y'' = -100y + 99\sin x,\ \ y(0) = 1,\ \ y'(0) = 11,\quad x\epsilon[0, 1000],$$

where the analytical solution is given by $y(x) = \cos(10x) + \sin(10x) + \sin x$. We choose $w = 10$.

For this example, the accuracy and efficiency of the TBNM are measured by the end-point global errors for different values of h and the corresponding FNCs used. The TBNM is of fourth order and hence comparable with the fourth order exponential-fitted method given in [26]. It is observed from Table 1 that the results produced by the TBNM are better than those given in [26]. We also compare the computational efficiency of the two methods by considering the FNCs per integration step for each method. The TBNM requires only one function evaluation per step compared with four function evaluations per step for the method in [26]. Thus, for this example, the TBNM is clearly superior.

Example 5.2 We consider the IVP (see [30])

$$y'' + K^2 y = K^2 x,\ \ y(0) = 10^{-5},\ \ y'(0) = 1 - K10^{-5}\cot K,\quad x\epsilon[0, 100],$$

where the analytical solution is given by $y(x) = x + 10^{-5}(\cos(Kx) - cotK\sin(Kx))$, K = 314.16, and we choose $w = 314.16$.

This problem was chosen to demonstrate the performance of the TBNM on a highly oscillatory problem. The results obtained using the TBNM are displayed in Table 2 and compare with the Dissipative Chebyshev exponential-fitted method (CHEBY24) given in [30]. We note that although the CHEBY24

Table 1 A comparison of end point global errors for Example 5.1

N	h	TBNM(FNCs)	Simos(FNCs)
1000	1	3.3×10^{-2}(1001)	1.4×10^{-1}(4000)
2000	0.5	2.1×10^{-3}(2001)	3.5×10^{-2}(8000)
4000	0.25	6.0×10^{-5}(4001)	1.1×10^{-3}(12000)
8000	0.125	2.1×10^{-5}(8001)	8.4×10^{-5}(32000)
16000	0.0625	1.3×10^{-6}(16001)	5.5×10^{-6}(64000)
32000	0.03125	7.8×10^{-8}(32001)	3.5×10^{-7}(128000)

Table 2 Results for Example 5.2

Method	N	FNCs	Absolute error
CHEBY24	9	450	1.84×10^{-11}
TBNM	256	257	9.35×10^{-12}

uses fewer steps, the TBNM uses fewer number of FNCs with better accuracy. Hence, the TBNM is very competitive with the method given in [30]. Details of the numerical results are given in Table 2.

Example 5.3 We consider the nonlinear Duffing equation which was also solved by Simos [26] and Ixaru and Berghe [12] on [0, 300]

$$y'' + y + y^3 = B\cos\Omega x,\ \ y(0) = C_0,\ y'(0) = 0.$$

The analytical solution is given by

$$y(x) = C_1\cos(\Omega x) + C_2\cos(3\Omega x) + C_3\cos(5\Omega x) + C_4\cos(7\Omega x),$$

where $\Omega = 1.01$, $C_0 = 0.200426728069$, $C_1 = 0.200179477536$, $C_2 = 0.246946143 \times 10^{-3}$, $C_3 = 0.304016 \times 10^{-6}$, $C_4 = 0.374 \times 10^{-9}$. We choose $w = 1.01$.

For this example, the end-point global errors for TBNM is compared with the methods given in [12, 26], since all the methods are of fourth order. It is observed from Table 3 that the results produced by the TBNM are better than those given in [26] and highly competitive with the method given in [12]. The TBNM requires only one function evaluation per step compared with four function evaluations per step for the methods in [12, 26]. Hence the TBNM is more efficient.

Example 5.4 We consider the nonlinear system of second order IVP (see [8])

$$y_1'' = (y_1 - y_2)^3 + 6368y_1 - 6384y_2 + 42\cos(10x),\quad y_1(0) = 0.5,\ y_1'(0) = 0,$$
$$y_1'' = -(y_1 - y_2)^3 + 12768y_1 - 12784y_2 + 42\cos(10x),\quad y_2(0) = 0.5,\ y_2'(0) = 0,$$
$$x\epsilon[0, 10],$$

with exact solution $y_1(x) = y_2(x) = \cos(4x) - \cos(4x)/2$.

This problem was chosen to demonstrate the performance of the TBNM on a nonlinear system. The accuracy and efficiency of the TBNM are measured by the end-point global errors for the y-component and the corresponding FNCs used. The results obtained using the TBNM are displayed in Table 4 and

Table 3 A comparison of end point global errors for Example 5.3

h	TBNM	Simos	Ixaru
1	1.31×10^{-3}	1.70×10^{-3}	1.10×10^{-3}
0.5	7.53×10^{-5}	1.88×10^{-4}	5.42×10^{-5}
0.25	2.47×10^{-6}	1.37×10^{-5}	1.86×10^{-6}
0.125	1.34×10^{-7}	8.70×10^{-7}	6.19×10^{-8}
0.0625	8.10×10^{-9}	5.41×10^{-8}	2.40×10^{-9}

Table 4 The correct decimal digit at the endpoint for Example 5.4

TIRK3		RADAU5		EFRK43		TBNM	
FNCs	Err	FNCs	Err	FNCs	Err	FNCs	Err
907	2.5×10^{-4}	853	2.2×10^{-4}	2057	3.7×10^{-4}	602	2.1×10^{-4}
1288	6.6×10^{-6}	1208	4.4×10^{-4}	1715	3.0×10^{-4}	1202	1.3×10^{-5}
1682	7.0×10^{-6}	1639	6.0×10^{-6}	3079	2.7×10^{-5}	1602	4.1×10^{-6}

compare with those given in [8]. It is seen from Table 4 that TBNM performs generally better than those in [8] in terms of accuracy and efficiency.

Example 5.5 We consider the stiff second order IVP (see [1])

$$y_1'' = (\varepsilon - 2)y_1 + (2\varepsilon - 2)y_2, \quad y_2'' = (1 - \varepsilon)y_1 + (1 - 2\varepsilon)y_2,$$

$$y_1(0) = 2, \quad y_1'(0) = 0, \quad y_2(0) = -1, \quad y_2'(0) = 0, \quad \varepsilon = 2500, \quad x \epsilon [0, 10\pi].$$

$$y_1(x) = 2\cos x, \quad y_2(x) = -\cos x, \text{where } \varepsilon \text{ is an arbitrary parameter and } w = 1.$$

This problem was chosen to justify the stability of the TBNM. The eigenvalues of the matrix of coefficients of the the equations for y_1'' and y_2'' are -1 and $-\varepsilon$, thus, the analytical solution of the system exhibit two frequencies 1 and $\sqrt{\varepsilon}$, however the initial conditions eliminate the high frequency component $\sqrt{\varepsilon}$ (see [1]). The method is stable when $q \in [0, 12]$. In Table 5, we give the absolute errors at selected values of x, which indicate that choosing $N = 454$, the method is stable since for this value of N, $q \in [0, 12]$. However, for $N = 452$, $q \ni [0, 12]$, hence the method becomes unstable.

Example 5.6 We consider the two-body problem which was also solved by Ozawa [21] on $[0, 50\pi]$.

$$y_1'' = -\frac{y_1}{r^3}, \quad y_2'' = -\frac{y_2}{r^3}, \quad r = \sqrt{y_1^2 + y_2^2},$$

$$y_1(0) = 1 - e, \quad y_1'(0) = 0, \quad y_2(0) = 0, \quad y_2'(0) = \sqrt{\frac{1+e}{1-e}}.$$

where e $(0 \le e < 1)$ is an eccentricity. The exact solution of this problem is $y_1(x) = \cos(\kappa) - e$, $\quad y_2(x) = \sqrt{1 - e^2}\sin(\kappa)$, where κ is the solution of the Kepler's equation $\kappa = x + e\sin(\kappa)$. We choose $w = 1$.

Table 5 Results for Example 5.6

x	$N = 454(q \in [0, 12])$ Err	$N = 452(q \ni [0, 12])$ Err
$\frac{5\pi}{2}$	4.34×10^{-11}	3.43×10^{-11}
5π	4.66×10^{-12}	6.93×10^{-8}
$\frac{15\pi}{2}$	1.30×10^{-10}	3.06×10^{-4}
10π	1.01×10^{-11}	5.41×10^{0}

Table 6 The endpoint errors and steps at $x = 50\pi$ with $e = 0.005$ for Example 5.6

TBNM		FESDIRK4(3)		ESDIRK4(3)	
Steps	Error	Steps	Error	Steps	Error
100	2.907×10^{0}	170	2.866×10^{-1}	277	2.153×10^{0}
200	6.630×10^{-3}	225	7.846×10^{-3}	496	1.494×10^{-1}
400	9.869×10^{-4}	381	1.399×10^{-3}	884	9.359×10^{-3}
800	4.870×10^{-6}	680	1.690×10^{-4}	1573	6.200×10^{-4}
1200	1.967×10^{-7}	1207	1.846×10^{-5}	2796	4.416×10^{-5}
1600	2.001×10^{-8}	2144	1.938×10^{-6}	4970	3.412×10^{-6}
2000	3.618×10^{-9}	3806	1.993×10^{-7}	8833	2.848×10^{-7}
3200	7.284×10^{-10}	6762	2.021×10^{-8}	15706	2.530×10^{-8}

The results obtained using the TBNM are displayed in Table 6 and compare with the explicit singly diagonally implicit Runge-Kutta (ESDIRK) and the functionally fitted ESDIRK (FESDIRK) methods given in [21]. It is seen from Table 6 that TBNM performs generally better than those in [8] in terms of accuracy(smaller steps) and efficiency (smaller FNCs).

5.2 Problems where y' appears explicitly

In this subsection, we show that the TBNM applicable to problems where y' appears explicitly.

Example 5.7 We consider the given Bessel's IVP solved on [1, 8] (see [29]).

$$x^2y'' + xy' + \left(x^2 - 0.25\right)y = 0, \quad y(1) = \sqrt{\frac{2}{\pi}}\sin 1 \simeq 0.6713967071418031,$$

$$y'(1) = (2\cos 1 - \sin 1)/\sqrt{2\pi} \simeq 0.0954005144474746.$$

$$\text{Exact}: y(x) = J_{1/2}(x) = \sqrt{\frac{2}{\pi x}}\sin x$$

The theoretical solution at $x = 8$ is $y(8) = \sqrt{\frac{2}{8\pi}}\sin(8) \simeq 0.279092789108058969$. We choose $w = \|\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\|$, evaluated at (x_0, y_0, y_0').

This problem was chosen to demonstrate the performance of the TBNM on a general second order IVP with variable coefficients. It was solved using

Table 7 The absolute errors at the endpoint for Example 5.7

	RK4		TBNM	
N	FNCs	Err	FNCs	Err
8	64	5.7×10^{-4}	9	5.1×10^{-4}
16	128	2.2×10^{-4}	17	7.3×10^{-5}
32	256	1.8×10^{-5}	33	5.8×10^{-6}
64	512	1.3×10^{-6}	65	3.9×10^{-7}
128	1024	8.4×10^{-8}	129	2.5×10^{-8}

the fourth-order Runge-Kutta method (RK4) and TBNM. We have chosen to compare these methods because their orders are the same. The absolute error for the y-component at the endpoint is given in Table 7. It is obvious from Table 7 that TBNM performs better than the RK4 method in terms of accuracy (smaller errors) and is more efficient (smaller FNCs). We note that we did not find any exponentially fitted method that has been used to solve this problem.

6 Conclusion

We have proposed a TBNM for $y'' = f(x, y, y')$. The method has order 4, is self-starting, provides good accuracy, and requires only one function evaluation per integration step. Numerical experiments performed using the TBNM show that the method is accurate and efficient. Our future research will be focused on developing a strategy for calculating the optimum frequency for the TBNM and implementing a variable step version of the method.

Acknowledgements The authors are very grateful to Professor Higinio Ramos of the Scientific Computing Group, Universidad de Salamanca, Spain for his several valuable suggestions that greatly improved the manuscript.

References

1. D'Ambrosio, R., Ferro, M., Paternoster, B.: Two-step hybrid collocation methods for $y'' = f(x, y)$. Appl. Math. Lett. **22**, 1076–1080 (2009)
2. Awoyemi, D.O.: A new sixth-order algorithm for general second order ordinary differential equation. Int. J. Comput. Math. **77**, 117–124 (2001)
3. Brugnano, L., Trigiante, D.: Solving Differential Problems by Multitep Initial and Boundary Value Methods. Gordon and Breach Science Publishers, Amsterdam (1998)
4. Butcher, J.C.: The Numerical Analysis of Ordinary Differential Equations, Runge-Kutta and General Linear Methods. Wiley, New York (1987)
5. Coleman, J.P., Duxbury, S.C.: Mixed collocation methods for $y'' = f(x, y)$. J. Comput. Appl. Math. **126**, 47–75 (2000)
6. Coleman, J.P., GR. Ixaru, G.: P-stability and exponential-fitting methods for $y'' = f(x, y)$. IMA J. Numer. Anal. **16**, 179–199 (1996)
7. Fatunla, S.O.: Block methods for second order IVPs. Int. J. Comput. Math. **41**(9), 55–63 (1991)
8. Nguyen, H.S., Sidje, R.B., Cong, N.H.: Analysis of trigonometric implicit Runge-Kutta methods. J. Comput. Appl. Math. **198**, 187–207 (2007)
9. Franco, J.M.: Runge-Kutta-Nyström methods adapted to the numerical integration of perturbed oscillators. Comput. Phys. Commun. **147**, 770–787 (2002)
10. Hairer, E.: Méthodes de Nyström pour l'équation différentielle $y'' = f(x, y)$. Numer. Math. **25**, 283–300 (1977)
11. Hairer, E., Wanner, G.: A Theory for Nystrom methods. Numer. Math. **25**, 383–400 (1976)
12. Ixaru, L., Berghe, G.V.: Exponential Fitting. Kluwer, Dordrecht, Netherlands (2004)
13. Jator, S.N.: Solving second order initial value problems by a hybrid multistep method without predictors. Appl. Math. Comput. **217**, 4036–4046 (2010)
14. Jator, S.N., Li, J.: A self-starting linear multistep method for a direct solution of the general second order initial value problem. Int. J. Comput. Math. **86**(5), 827–836 (2009)
15. Jator, S.N.: A sixth order linear multistep method for the direct solution of $y'' = f(x, y, y')$. Int. J. Pure Appl. Math. **40**, 457–472 (2007)

16. Keiper, J.B., Gear, C.W.: The analysis of generalized backwards difference formula methods applied to Hessenberg form differential-algebraic equations. SIAM J. Numer. Anal. **28**, 833–858 (1991)
17. Lambert, J.D., Watson, A.: Symmetric multistep method for periodic initial value problem. J. Inst. Math. Appl. **18**, 189–202 (1976)
18. Lambert, J.D.: Computational Methods in Ordinary Differential Equations. John Wiley, New York (1973)
19. Milne, W.E.: Numerical Solution of Differential Equations. John Wiley and Sons (1953)
20. Onumanyi, P., Sirisena, U.W., Jator, S.N.: Continuous finite difference approximations for solving differential equations. Int. J. Comput. Math. **72**, 15–27 (1999)
21. Ozawa, K.: A functionally fitted three-stage explicit singly diagonally implicit Runge-Kutta method. Japan J. Indust. Appl. Math. **22**, 403–427 (2005)
22. Ramos, H., Vigo-Aguiar, J.: Variable stepsize Stôrmer-Cowell methods. Math. Comput. Model. **42**, 837–846 (2005)
23. Rosser, J.D.: A Runge-kutta for all seasons. SIAM Rev. **9**, 417–452 (1967)
24. Shampine, L.F., Watts, H.A.: Block implicit one-step methods. Math. Comput. **23**, 731–740 (1969)
25. Sommeijer, B.P.: Explicit, high-order Runge-Kutta-Nyström methods for parallel computers. Appl. Numer. Math. **13**, 221–240 (1993)
26. Simos, T.E.: An exponentially-fitted Runge-Kutta method for the numerical integration of initial-value problems with periodic or oscillating solutions. Comput. Phys. Commun. **115**, 1–8 (1998)
27. Stiefel, E., Bettis, D.G.: Stabilization of Cowell's method. Numer. Math. **13**, 154–175 (1969)
28. Tsitouras, Ch.: Explicit eighth order two-step method with nine stages for integrating oscillatory problems. Int. J. Modern Phys. C **17**, 861–876 (2006)
29. Vigo-Aguiar, J., Ramos, H.: Variable stepsize implementation of multistep methods for $y'' = f(x, y, y')$. J. Comput. Appl. Math. **192**, 114–131 (2006)
30. Vigo-Aguiar, J., Ramos, H.: Dissipative Chebyshev exponential-fitted methods for numerical solution of second-order differential equations. J. Comput. Appl. Math. **158**, 187–211 (2003)