IJAMS Peer-reviewed journal
International Journal of Applied Mathematics and Simulation

An effective operational matrix method for the solution of non-linear third-order initial value problems

Abdelkader Laiadi[1] https://orcid.org/0000-0003-1034-4408.

DOI: https://doi.org/10.69717/ijams.v2.i1.118

abstract

The present paper provides a new technique using the clique polynomials as basis function for the operational matrices to obtain numerical solutions of third-order non-linear ordinary differential equations. It aims to find all solutions as easy as possible. Numerical results derived using the proposed techniques are compared with the exact solution or the solutions obtained by other existing methods. The new numerical examples were examined to show that the new approach is highly efficient and accurate. The approximate solutions can be very easily calculated using computer program Matlab.

keywords

The clique polynomials; Operational matrix; Collocation points; Third-order of differential equations; Initial value problems.

2020 Mathematics Subject Classification

Primary 62G05; 62G07 · Secondary 62R10. ·

1. Introduction

Many problems in physics, chemistry, and engineering science are modeled as third-order boundary value problems or initial value problems. These boundary value problems can be found in different areas of applied mathematics and physics such as, in the deflection of a curved beam having a constant, thin-film flow, and gravity-driven flows (see Momoniat and Mahomed, 2010; Tuck and Schwartz, 1990). Most nonlinear differential equations do not have exact solutions, so approximation and numerical techniques must be used. Many researchers developed some methods to solve boundary and initial value problems of different order such as Agarwal, 1986; Butcher, 2016; Fatima, 2024 and others. In this paper, we focus on initial value problems of third-order nonlinear ordinary differential equations.

$$\begin{cases} y''' = f\left(x, y\left(x\right), y'\left(x\right), y''\left(x\right)\right) \\ y\left(x_0\right) = \alpha, y'\left(x_0\right) = \beta, y''\left(x_0\right) = \gamma, x \in [x_0, x_{end}] \end{cases} \tag{1}$$

where $y(x) \in R, f := R \times R \times R \times R \to R$ is a continuous function and α, β and γ are constants. Several direct methods are widely proposed by researchers in solving third-order differential equations such as iterative method, Traub's method Chun and Kim, 2010, block method Abu Arqub et al., 2013; Mehrkanoon, 2011; Yap et al., 2014, Runge-Kutta method Fang et al., 2014; Lee et al., 2020; You and Chen, 2013, operational matrices of Bernstein polynomials method Khataybeh et al., 2019; Malik et al., 2021 and more.
The main of this paper is to apply the new operational matrix of integration method using clique polynomials to solve the third-order initial value problems. It is shown that the method provides the solution in a rapid

[1]Department of Mathematics, University of Biskra. 07000, Biskra, Algeria , abdelkader.laiadi@univ-biskra.dz

Communicated Editor: M. Berbiche
Manuscript received *Oct 27, 2024;* ***revised*** *April 24, 2025;* ***accepted*** *May 11, 2025;* ***published on line*** *June 14, 2025.*

convergent series. The other operational matrix method using clique polynomials has been used by Kumbinarasaiah et al., 2021 and Ganji et al., 2021 to solve effectively the non linear Klein Gordon equation and non-linear fractional Klein Gordon equation, which converge rapidly to accurate solutions. We show that the initial value problems of third-order can be solved efficiently using the clique polynomials. The present method converts Eq. (1) to a system of algebraic equations which can be solved easily. The capability of the method shall be tested on a linear and nonlinear third-order differential equations.
This paper is arranged as follows. In Section 2, we give the interesting properties of clique polynomials and there convergence analysis. In Section 3, we construct the operational matrix technique using the clique polynomials for solving numerically the nonlinear third-order differential equations. Section 4 includes to present several results and discussions to show the efficiency and simplicity of the proposed method. Finally, conclusion is given in Section 5.

2. Clique polynomials and convergence analysis

Let G be a graph that is free from multi edges and loops. The clique polynomial of a graph G, denoted by $C(G;x)$, is characterized by Hoede and Li, 1994

$$C(G;x)=\sum_{k=0}^{n} a_k x^k$$

where a_k represent the total distinct $k-$cliques in graph of size k, with $a_0=1$. The clique polynomial of a complete graph K_n with $n-$vertices is given by

$$C\left(K_n;x\right)=(1+x)^n=\sum_{k=0}^{n}\binom{n}{k}x^k$$

where $\binom{n}{k}=\frac{n!}{k!(n-k)!}$
In particular

$$\begin{aligned} C(K_0;x) &= 1 \\ C(K_1;x) &= 1+x \\ C(K_2;x) &= 1+2x+x^2 \\ C(K_3;x) &= 1+3x+3x^2+x^3 \end{aligned}$$

Let $B=\{C_n(x)=C(K_n,x), n\in N\}$. Clearly B is Banach space on closed subset A of R with norm given by

$$\|C_n\|=\sup_{\forall x\in A}|C_n(x)| \qquad \forall C_n\in B(A)$$

We can approximate any function $f(x)$ in $L^2[0,1]$ in terms of the clique polynomial as (see Ganji et al., 2021; Kumbinarasaiah et al., 2021)

$$f(x)\approx \widetilde{f}(x)=\sum_{i=0}^{n-1} a_i C\left(K_i;x\right)$$

We can write

$$f(x)=\sum_{i=0}^{n-1} a_i\left(\sum_{k=0}^{i}\binom{i}{k}x^k\right)=A^T P X(x)$$

where $A^T=[a_0,a_1,\ldots,a_{n-1}]$, $X(x)=[1,x,\ldots,x^{n-1}]^T$ and P is the lower triangular $n\times n$ matrices defined by

$$p_{ij}=\begin{cases} 0 & j>i, i,j=1,2,...,n \\ \frac{(i-1)!}{(i-j)!(j-1)} & i\geq j, i,j=1,2,...,n \end{cases}$$

3. Description of the clique polynomial operational matrix method

We consider the clique polynomial operational matrix method along with collocation points to solve the following third-order of differential equations

$$y^{(3)}=f(x,y,y',y''), 0\leq x\leq 1 \tag{2}$$

with the initial conditions

$$y(0) = b_1, y'(0) = b_2, y''(0) = b_3 \tag{3}$$

where b_1, b_2, b_3 are real constants and f is a given continuous on $[0,1]$, nonlinear function. We assume that

$$y'''(x) = A^T PX(x) \tag{4}$$

Where A is an unknown vector to be determined $A^T = [a_0, a_1, \dots, a_{n-1}]$, $X(x)$ is the known vector defined above and

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 2 & 1 & \ddots & \cdots & 0 \\ 1 & 3 & 3 & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 1 & n-1 & \frac{(n-1)(n-2)}{2!} & \cdots & n-1 & 1 \end{bmatrix}$$

For solving the equation (2), we calcul the derivatives $y^{(k)}(x)$ where $k = 0,1,2,3, x \in [0,1]$ and with the initial conditions (3)

It is easy to prove that this identity

$$\underbrace{\int_0^x \int_0^x \dots \int_0^x}_{k \text{ times}} A^T PX(t)\,dt = A^T PM_k x^k X(x)$$

where M_k is the $n \times n$ matrices

$$M_k = \begin{bmatrix} \frac{1}{k!} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{2\times 3\times \dots (k+1)} & 0 & \cdots & 0 \\ 0 & 0 & \frac{1}{3\times 4\times \dots (k+2)} & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \frac{1}{n(n+1)\dots(n+k-1)} \end{bmatrix}$$

Integrating equation (4) third times on bothside with respect to x limit between 0 and x, we obtain

$$y(x) = b_1 + b_2 x + \frac{b_3}{2}x^2 + \int_0^x \int_0^x \int_0^x A^T PX(t)dt$$

After integration yields

$$y(x) = b_1 + b_2 x + \frac{b_3}{2}x^2 + A^T PM_3 x^3 X(x)$$

where

$$M_3 = \begin{bmatrix} \frac{1}{3!} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{4!} & 0 & \cdots & 0 \\ 0 & 0 & \frac{1}{3\times 4\times 5} & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \frac{1}{n(n+1)(n+2)} \end{bmatrix}$$

Now by substituting y, y', y'', y''' into equation (2) and collocate this equation by the following collocation points $x_i = \frac{2i-1}{2n}, i = 1, ..., n$, we get a system of n non linear equations with n unknowns $(a_0, a_1, ..., a_{n-1})$. The unknown coefficients are determined by satisfying the remaining the initial conditions (3) at chosen collocation points. This system can be solved by using the Newton method.

4. Numerical results

In order to test the proposed method, we present some numerical results obtained by applying operational matrix method to find numerical approximations of the solutions of some test problems $\left(x_i = \frac{1}{10}i; i = 0, 1, ..., 10\right)$. We will discuss the new numerical examples of third-order initial value problems. The tables 1-4 clearly show the improvements we achieved if compared to the exact solution. Figures 1, 3 and 5 show the comparison between the numerical solutions and the exact solutions of the initial value problems (Examples 1-3). Examining these tables, it is clear that the absolute errors were seem to be small. It is should be noted that the

Table 1: Numerical results for Example 1 ($n = 10$)

x	Exact solution	Numerical solution	Errors
0	-1	-1	0
0.1	-0.994995834721974	-0.994995834723177	$1.20281562487889E-12$
0.2	-0.979933422158758	-0.979933422162680	$3.92197385679083E-12$
0.3	-0.954663510874394	-0.954663510881710	$7.31581462076747E-12$
0.4	-0.918939005997115	-0.918939006007955	$1.08402176124400E-11$
0.5	-0.872417438109627	-0.872417438122700	$1.30726540703563E-11$
0.6	-0.814664385090322	-0.814664385113334	$2.30125918321278E-11$
0.7	-0.745157812715512	-0.745157812779116	$6.36040109469604E-11$
0.8	-0.663293290652835	-0.663293290813850	$1.61015867305991E-10$
0.9	-0.568390031729336	-0.568390032047044	$3.17708304109487E-10$
1	-0.459697694131860	-0.459697694888935	$7.57074403168190E-10$

approximate solution approaches the exact solution as n, the number of the basis functions, increases. All numerical computations have been done in Matlab (see Matlab program below), the program execution time by this method is 47 second.
Where

$$\text{Absolute error} = |\text{Exact solution} - \text{Numerical solution}|$$

Example 1 Consider the linear third-order initial value problem

$$y''' = sin(x), 0 \leq x \leq 1 \tag{5}$$

with initial conditions

$$y(0) = -1, y'(0) = 0, y''(0) = 1 \tag{6}$$

The analytic solution of the above problem is

$$y = cos(x) + x^2 - 2 \tag{7}$$

We have

$$y(x) = -1 + \frac{1}{2}x^2 + A^T P M_3 x^3 X(x)$$

Substituting equation (4) into (5) yields

$$A^T P X(x) = sin(x)$$

We collocate this equation at the collocation points $x_i = \frac{2i-1}{2n}, i = 1, ..., n$ to obtain numerical values of y. By using the conditions (6), the obtained system is solved, yielding the following results for $n = 10$

$$A = \begin{bmatrix} -0.810695 \\ 0.332789 \\ 1.038764 \\ -1.157273 \\ 1.142639 \\ -0.856889 \\ 0.418821 \\ -0.129569 \\ 0.023261 \\ -0.001849 \end{bmatrix}$$

Table 1 and 2 show that the numerical solutions and the errors obtained for linear third-order initial value problem (5) (Example 1) by using the present method and compared with the exact solution (7) for $n = 10$ and $n = 15$ respectively. Figure 1 shows the comparison between the approximate solution and the exact solution (7) of the problem (5). In Figure 2, the absolute errors have been shown at distinct points.

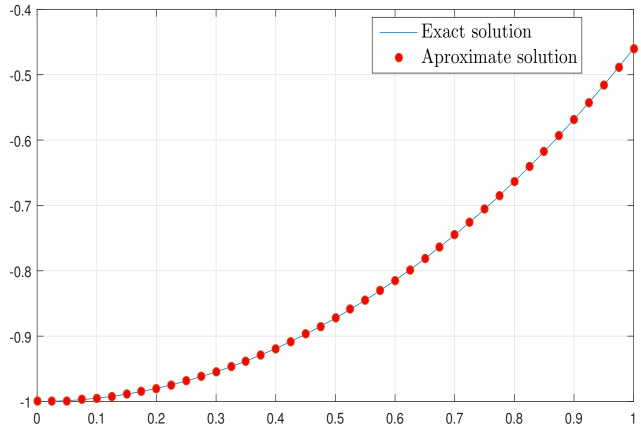

Figure 1: Comparison of approximate and exact solution for Example 1.

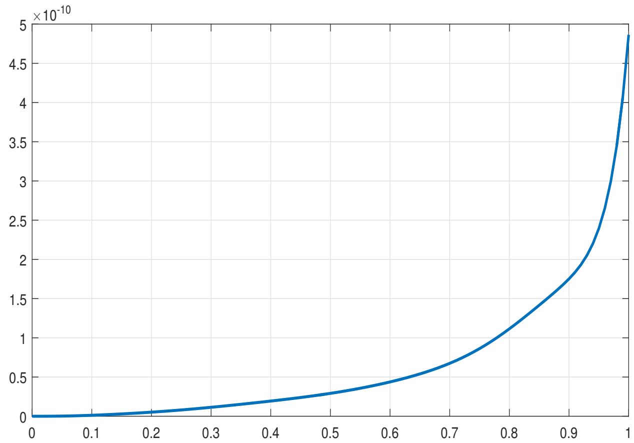

Figure 2: Error Analysis of Example 1.

Table 2: Numerical results for Example 1 ($n = 15$)

x	Exact solution	Numerical solution	Errors
0	-1	-1	0
0.1	-0.994995834721974	-0.994995834721283	$-9.7666319476275E-13$
0.2	-0.979933422158758	-0.979933422154582	$-6.9754202414174E-12$
0.3	-0.954663510874394	-0.954663510866042	$-2.57520671453904E-11$
0.4	-0.918939005997115	-0.918939005974870	$-6.47839559775321E-11$
0.5	-0.872417438109627	-0.872417438040750	$-1.19521503805231E-10$
0.6	-0.814664385090322	-0.814664384930306	$-1.78378645188104E-10$
0.7	-0.745157812715512	-0.745157812411764	$-2.44772868640553E-10$
0.8	-0.663293290652835	-0.663293290068459	$-3.49193451931740E-10$
0.9	-0.568390031729336	-0.568390030578339	$-5.29264299053978E-10$
1	-0.459697694131860	-0.459697691704301	$-7.88076048863218E-10$

Example 2 Consider the linear third-order initial value problem

$$y''' = 8e^{2x} + 2, 0 \leq x \leq 1 \tag{8}$$

with initial conditions

$$y\left(0\right) = -2, y'\left(0\right) = 2, y''\left(0\right) = 4 \tag{9}$$

The analytic solution of the above problem is

$$y\left(x\right) = e^{2x} + \frac{1}{3}x^3 - 3 \tag{10}$$

By solving the equation (8) with conditions (9) we obtain the vector A for $n = 10$

$$A = \begin{bmatrix} 3.099290 \\ 2.071009 \\ 2.360541 \\ 1.332371 \\ 0.476479 \\ 0.862816 \\ -0.466501 \\ 0.338103 \\ -0.090077 \\ 0.015966 \end{bmatrix}$$

Table 3 shows that the approximate solutions and the errors obtained for linear third-order initial value problem (8) (Example 2) and compared with the exact solution (10) for $n = 10$. Figure 3 shows the comparison between the approximate solution and the exact solution of the problem (8). Figure 4 shows the error Analysis of Example 2.

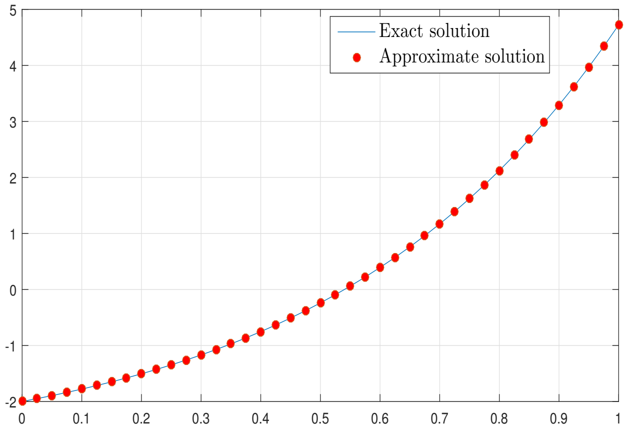

Figure 3: Comparison of approximate and exact solution for Example 2.

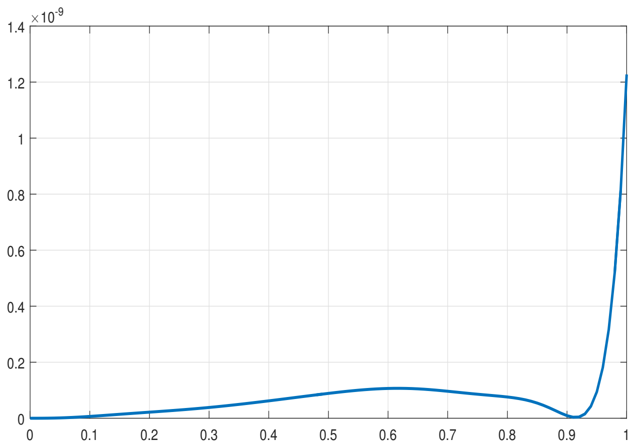

Figure 4: Error Analysis of Example 2.

Table 3: Numerical results for Example 2 ($n = 10$)

x	Exact solution	Numerical solution	Errors
0	-2	-2	0
0.1	-1.778263908506500	-1.77826390851309	$6.58939569575523E-12$
0.2	-1.505508635692060	-1.50550863571379	$2.17246221012601E-11$
0.3	-1.168881199609490	-1.16888119964800	$3.85114162781970E-11$
0.4	-0.753125738174199	-0.753125738236310	$6.21112050680495E-11$
0.5	-0.240051504874288	-0.240051504963057	$8.87692142015339E-11$
0.6	$+0.392116922736547$	$+0.392116922630138$	$1.06409547839803E-10$
0.7	$+1.169533300178010$	$+1.169533300081640$	$9.63700230727227E-11$
0.8	$+2.123699091061780$	$+2.123699090985880$	$7.59063922828318E-11$
0.9	$+3.292647464412950$	$+3.292647464403680$	$9.26281273905261E-12$
1	$+4.722389432263980$	$+4.722389431036580$	$1.22740306807145E-09$

Table 4: Numerical results for Example 3

x	Exact solution	Numerical solution for $n = 7$	Numerical solution for $n = 10$
0	1	1	1
0.1	0.90483741803590	0.90483741804721	0.90483741803568
0.2	0.81873075307798	0.81873075312857	0.81873075307623
0.3	0.74081822068171	0.74081822079209	0.74081822067633
0.4	0.67032004603563	0.67032004622430	0.67032004602522
0.5	0.60653065971263	0.60653065999590	0.60653065969547
0.6	0.54881163609402	0.54881163648641	0.54881163606573
0.7	0.49658530379141	0.49658530430750	0.49658530374543
0.8	0.44932896411722	0.44932896476990	0.44932896404636
0.9	0.40656965974059	0.40656966054202	0.40656965963669
1	0.36787944117144	0.36787944214112	0.36787944102411

Example 3 Consider the non-linear third-order initial value problem

$$y''' + y'' + y'y = -e^{-2x}, 0 \le x \le 1 \tag{11}$$

with initial conditions

$$y\left(0\right) = 1, y'\left(0\right) = -1, y''\left(0\right) = 1 \tag{12}$$

The analytic solution of the above problem is

$$y\left(x\right) = e^{-x} \tag{13}$$

By solving the equation (11) with conditions (12) we obtain the vector A for $n = 10$

$$A = \begin{bmatrix} -2.197399 \\ -0.840032 \\ 9.378083 \\ -18.330639 \\ 20.881442 \\ -15.525444 \\ 7.625636 \\ -2.391506 \\ 0.434777 \\ -0.034917 \end{bmatrix}$$

Table 4 and 5 show that the numerical solutions and the errors obtained for non-linear third-order initial value problem (11) (Example 3) and compared with the exact solution (13) for $n = 7$ and $n = 10$ respectively. Figure 5 shows the comparison between the approximate solution and the exact solution of the problem (11). Figure 6 shows the error Analysis of Example 3.

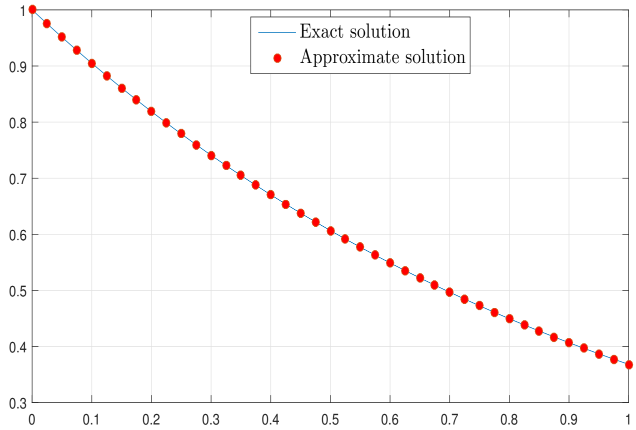

Figure 5: Comparison of approximate and exact solution for Example 3.

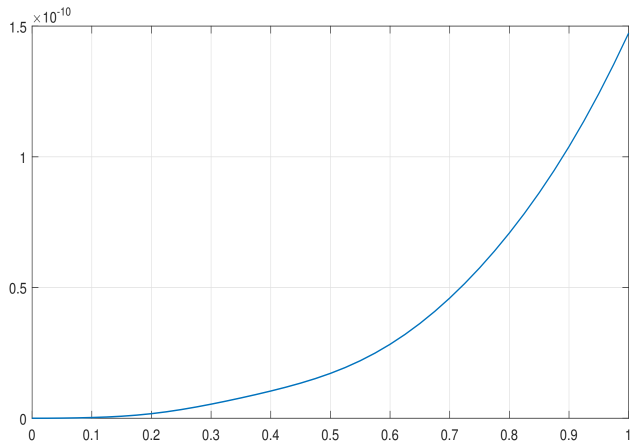

Figure 6: Error Analysis of Example 3.

Table 5: Errors obtained for Example 3

x	Errors for $n=7$	Errors for $n=10$
0	0	0
0.1	$1.12592157819336E-11$	$2.74891220897189E-13$
0.2	$5.05927522098659E-11$	$1.74982250911171E-12$
0.3	$1.10377484929813E-10$	$5.38702416008618E-12$
0.4	$1.88661086752973E-10$	$1.04127817479593E-11$
0.5	$2.83306489379243E-10$	$1.71629377376803E-11$
0.6	$3.92390342440763E-10$	$2.82878165336342E-11$
0.7	$5.16110376658219E-10$	$4.59714488698637E-11$
0.8	$6.52678022738939E-10$	$7.08584302344661E-11$
0.9	$8.01421373708422E-10$	$1.03907771276113E-10$
1	$9.69684443852259E-10$	$1.47327927635388E-10$

Table 6: Numerical results for Example 4 ($n=10$ and $B=1$)

x	Hb. method Adesanya et al., 2013	Bp. method Khataybeh et al., 2019	(CP) method
0.1	0.004999979166110	0.0049999583341723	0.004999958453095
0.2	0.019998666668590	0.0199986668419935	0.019998667405196
0.3	0.044998481293978	0.0449898794745896	0.044989880928476
0.4	0.079991467388617	0.0799573779857994	0.079957380252171
0.5	0.124967454367055	0.1248700575229549	0.124870060064380
0.6	0.179902837409194	0.1796771412454840	0.179677143791334
0.7	0.244755067600357	0.2443036169821510	0.244303617750305
0.8	0.319454500640289	0.3186460093102460	0.318646005190335
0.9	0.403894871267148	0.4025686205525250	0.402568610236483
1	0.497922483110430	0.4959003827831510	0.495900375094189

Example 4 Now consider the nonlinear boundary layer equation

$$2y''' + y''y = 0, 0 \leq x \leq 1 \tag{14}$$

with initial conditions

$$y(0) = 0, y'(0) = 0, y''(0) = B \tag{15}$$

This equation is famously known as the Blasius equation. The aim of solving Blasius equation to get the value $y''(0)$ to evaluate the shear stress at the plate. Blasius equation has been solved using different methods like series expansions, Runge Kutta, differential transformation and others. By solving the Equation (14) with conditions (15) we obtain the vector A for $n=10$ and $B=1$

$$A = \begin{bmatrix} -5.258858 \\ 34.500595 \\ -102.128598 \\ 176.867272 \\ -196.065300 \\ 143.930229 \\ -69.967095 \\ 21.716530 \\ -3.904633 \\ 0.309859 \end{bmatrix}$$

Table 6 show that the numerical solutions for non-linear Blasius equation (14) (Example 4) by using presented method ((CP) method) and compared with another numerical methods for $n=10$ and $B=1$ (Hb. method is Hybrid block method and Bp is Bernstein polynomials). In all the above the results, it is noticed that the numerical solutions achieved by our method coincide quite well with other methods available in the literature and signify that the proposed method is viable and convergent.

5. Conclusion

In this paper, we introduced an effective operational matrix method for solving nonlinear third-order of non-linear ordinary differential equations by constructing a new matrices using the clique polynomials. The proposed approach has been successfully applied to various numerical examples to demonstrate its applicability and accuracy. Numerical simulations confirm that the approximate solutions are in excellent agreement with solutions obtained by other existing methods or exact solution, and a highly accurate solution can be obtained in a few iterates, which is apparent through numerical results. The proposed algorithm is an efficient and highly promising technique for solving third-order non-linear ordinary differential equations. The method might be applied for a system of differential equations or higher order of boundary value problems.

Acknowledgment

The author would like to thank to express their gratitude to the anonymous reviewers for their constructive feedback and the editor for their thorough evaluation of this manuscript.

Declarations

The author declares that he has no conflicts of interest.

References

Abu Arqub, O., Abo-Hammour, Z., Al-Badarneh, R., & Momani, S. (2013). A reliable analytical method for solving higher-order initial value problems. *Discrete Dynamics in Nature and Society*, *2013*(1), 1–12. https://doi.org/10.1155/2013/673829

Adesanya, A. O., Udoh, D. M., & Ajileye, A. M. (2013). A new hybrid block method for the solution of general third order initial value problems of ordinary differential equations. *International journal of pure and applied mathematics*, *86*(02), 365–375. https://doi.org/10.12732/ijpam.v86i2.11

Agarwal, R. P. (1986). *Boundary value problems from higher order differential equations*. World Scientific. https://doi.org/10.1142/0266

Butcher, J. C. (2016). *Numerical methods for ordinary differential equations*. John Wiley & Sons. https://doi.org/10.1002/9780470753767

Chun, C., & Kim, Y. I. (2010). Several new third-order iterative methods for solving nonlinear equations. *Acta applicandae mathematicae*, *109*(1), 1053–1063. https://doi.org/10.1007/s10440-008-9359-3

Fang, Y. L., You, X., & Ming, Q. (2014). Trigonometrically fitted two derivative runge-kutta methods for solving oscillatory differential equations. *Numer. Algorithms*, *65*, 651–667. https://doi.org/10.1007/s11075-013-9802-z

Fatima, O. (2024). Bat algorithm for solving ivps of current expression in series rl circuit constant voltage case. *The International Journal of Applied Mathematics and Simulation (IJAMS)*, *1*(2), 2992–1732. https://doi.org/10.69717/ijams.v1.i2.101

Ganji, R. M., Jafari, H., Kgarose, M., & Mohammadi, A. (2021). Numerical solutions of time-fractional klein-gordon equations by clique polynomials. *Alexandria Engineering Journal*, *60*, 4563–4571. https://doi.org/10.1016/j.aej.2021.03.026

Hoede, C., & Li, X. (1994). Clique polynomials and independent set polynomials of graphs. *Discrete Mathematics*, *125*(1-3), 219–228. https://doi.org/10.1016/0012-365X(94)90163-5

Khataybeh, S. A. N., Hashim, I., & Alshbool, M. (2019). Solving directly third-order odes using operational matrices of bernstein polynomials method with applications to fluid flow equations. *Journal of King Saud University-Science*, *31*(4), 822–826. https://doi.org/10.1016/j.jksus.2018.05.002

Kumbinarasaiah, S., Ramane, H. S., Pise, K. S., & Hariharan, G. (2021). Numerical-solution-for-nonlinear-klein–gordon equation via operational-matrix by clique polynomial of complete graphs. *International Journal of Applied and Computational Mathematics*, *07*(1), 1–19. https://doi.org/10.1007/s40819-020-00943-x

Lee, K. C., Senu, N., Ahmadian, A., Ibrahim, S. N. I., & Baleanu, D. (2020). Numerical study of third-order ordinary differential equations using a new class of two derivative runge-kutta type methods. *Alexandria Engineering Journal*, *59*(4), 2449–2467. https://doi.org/10.1016/j.aej.2020.03.008

Malik, R., Khan, F., Basit, M., Ghaffar, A., Sooppy Nisar, K., Mahmoud, E. E., & Lotayif, M. S. M. (2021). Bernstein basis functions based algorithm for solving system of third-order initial value problems. *Journal of Alexandria Engineering*, *60*(2), 2395–2404. https://doi.org/10.1016/j.aej.2020.12.036

Mehrkanoon, S. (2011). A direct variable step block multistep method for solving general third-order odes. *Numerical Algorithms*, *57*(1), 53–66. https://doi.org/10.1007/s11075-010-9413-x

Momoniat, E., & Mahomed, F. M. (2010). Symmetry reduction and numerical solution of a third-order ode from thin film flow. *Mathematical and Computational Applications*, *15*(4), 709–719. https://doi.org/10.3390/mca15040709

Tuck, E. O., & Schwartz, L. W. (1990). A numerical and asymptotic study of some third-order ordinary differential equations relevant to draining and coating flows. *SIAM review*, *32*(3), 453–469. https://doi.org/10.1137/1032079

Yap, L. K., Ismail, F., & Senu, N. (2014). An accurate block hybrid collocation method for third order ordinary differential equations. *Journal of Applied Mathematics*, *2014*(1), 549597. https://doi.org/10.1155/2014/549597

You, X., & Chen, Z. (2013). Direct integrators of runge–kutta type for special third-order ordinary differential equations. *Applied Numerical Mathematics*, *74*(1), 128–150. https://doi.org/10.1016/j.apnum.2013.07.005