FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 10 No. 24, November, 2026, pp 387 – 393
DOI: https://doi.org/10.33003/fjs-2026-1024-5693

A Chebyshev-Based Four-Step Hybrid Block Method for Second-Order Initial Value Problems

***[1]Hassan Mashood and [2]Mark I. Modebei**

[1]Department of Mathematics, University of Abuja, Abuja, Nigeria
[2]Mathematics Programme, National Mathematical Centre, Abuja, Nigeria.

*Corresponding authors' email: hassanmashood4@gmail.com

Received: 31 October 2026
Accepted: 15 November 2026
Published: 19 November, 2026

ABSTRACT
A four-step hybrid block method for second-order initial-value problems is obtained from a degree-ten Chebyshev approximation collocated at nine Chebyshev–Lobatto abscissae. The nine implicit stage equations and eight fully expanded block-output formulas are derived, with exact coefficient definitions. Polynomial moment analysis establishes uniform ninth-order block accuracy; consistency, zero-stability and conditional convergence are examined. The boundary-locus formulation of a two-parameter amplification matrix provides a graphical linear-stability diagnostic. Five exact-solution examples include harmonic and nonlinear Duffing oscillators, nonlinear velocity dependence, changing frequency, and an excited stiff transient. Comparisons with a published sixth-order Falkner hybrid block method, fourth-order Runge–Kutta and tenth-order Gauss collocation show smaller errors than the Falkner scheme on the selected meshes, but higher acceleration-evaluation counts. Test-problem origins and manufactured examples are identified separately, and the evidence does not imply universal computational superiority.

Keywords: Second-Order Initial-Value Problems, Chebyshev–Lobatto Collocation, Block Method, Zero-Stability, Nonlinear Numerical Experiments

INTRODUCTION

Second-order initial value problems arise when acceleration depends on the current state, time and velocity. A mass–spring–damper system, for example, obeys a balance between inertia, restoring force and dissipation. Related models describe orbital oscillation, vibration control and transient electrical or mechanical responses. We consider

$$y''(x) = f(x, y(x), y'(x)), \qquad y(a) = y_0, \quad y'(a) = d_0, \qquad a \le x \le b \tag{1}$$

Here x is the independent variable; y, y' and y'' denote displacement, velocity and acceleration, respectively; and f specifies the governing dynamics. Under appropriate continuity and local Lipschitz assumptions, (1) has a locally unique solution. Nonlinearity, rapid transients and oscillation frequently make analytic evaluation impractical. Direct block integration is therefore attractive because it approximates displacement and velocity without first replacing the equation by a larger first-order system.

The consistency and convergence theory of numerical integration has been developed by Henrici (1962) and Fatunla (1991). Shampine and Watts (1969) introduced implicit block implementations, while Lambert and Watson (1976) showed why periodic stability matters for direct second-order schemes. Jator and Li (2009) constructed a self-starting multistep formula. Jator and Lee (2014) subsequently highlighted the cost differences between block and predictor–corrector implementations. These studies underline an important point: algebraic order alone is insufficient to establish computational efficiency.

Modebei and Adeniyi (2019) used Chebyshev-based collocation for a six-step block unification method addressing fourth-order boundary-value problems. Although its differential order and boundary conditions differ from (1), the work illustrates the flexibility of continuous polynomial representations. Ramos, Jator and Modebei (2020) examined efficient k-step linear block methods for second-order initial-value problems and emphasised that implementation structure affects computational cost. Jator, Ramos and Modebei (2023) introduced a variable-step hybrid block method of order fourteen, providing a useful high-order reference point for direct integration. Taken together, these contributions motivate a careful study of block coefficients, node choices, stability and measurable numerical work.

Okuonghae and Ozobokeme (2024) presented a sixth-order Falkner hybrid block method with an explicitly published scheme. Their method is adopted as the principal published-method comparator here. Nonlinear second-order block integration has also attracted recent attention (Ali et al., 2026). For transparent testing, we combine standard oscillatory models with analytically constructed exact-solution problems. The latter follow the verification principle of the method of manufactured solutions (Roache, 2002); the individual origins are stated in Section 4.

Let$T_0(z) = 1$, $T_1(z) = z$ and $T_{j+1}(z) = 2zT_j(z) - T_{j-1}(z)$ for$j \ge 1$. The shifted Chebyshev basis $T_j^*(\tau) = T_j(\tau/2 - 1)$ spans the same polynomial space as the monomial basis (Mason & Handscomb, 2003). It can improve coefficient generation numerically, but does not by itself increase algebraic order. The present work derives an implicit four-step block scheme with nine Chebyshev–Lobatto abscissae and examines its order, stability and comparative accuracy on five independently checkable problems.

MATERIALS AND METHODS

Derivation of the Method

Consider (1) on the uniform partition$x_n = a + nh$, $n = 0,1, \dots, N$, where $h = (b-a)/N > 0$ and N is divisible by four. A single block occupies$[x_n, x_{n+4}]$, has length$H = 4h$, and is parameterised by$\tau = (x - x_n)/h \in [0,4]$. The desired approximation is a polynomial P_n of degree ten satisfying.

$P_n(0) = y_n, \quad P_n'(0) = hy_n', \quad P_n''(c_j) = h^2 f\left(x_n + hc_j, P_n(c_j), h^{-1}P_n'(c_j)\right), \quad j = 0, \dots, 8$ (2)

The nine collocation abscissae and cardinal polynomials are

$c_j = 2\left(1 - \cos\frac{j\pi}{8}\right), \qquad \ell_j(\tau) = \prod_{\substack{m=0 \\ m\neq j}}^{8} \frac{\tau - c_m}{c_j - c_m}, \qquad j = 0, \dots, 8$ (3)

Thus $c = \left(0, 2-\sqrt{2+\sqrt{2}}, 2-\sqrt{2}, 2-\sqrt{2-\sqrt{2}}, 2, 2+\sqrt{2-\sqrt{2}}, 2+\sqrt{2}, 2+\sqrt{2+\sqrt{2}}, 4\right)^T$. The polynomial is equivalently expressible as $P_n(\tau) = \sum_{r=0}^{10} a_r T_r^*(\tau)$. With $F_j = f\left(x_n + hc_j, Y_j, D_j\right)$, two integrations of $P_n''(\tau) = h^2 \sum_{j=0}^{8} \ell_j(\tau) F_j$ give

$P_n(\tau) = y_n + h\tau y_n' + h^2 \sum_{j=0}^{8} \beta_j(\tau) F_j, h^{-1}P_n'(\tau) = y_n' + h\sum_{j=0}^{8} \gamma_j(\tau) F_j$ (4)

$\beta_j(\tau) = \int_0^\tau (\tau - s)\,\ell_j(s)\,ds, \gamma_j(\tau) = \int_0^\tau \ell_j(s)\,ds.$ (5)

The complete implicit stage equations, including all fractional collocation locations, are

$$\begin{aligned} Y_j &= y_n + hc_j y_n' + h^2 \sum_{m=0}^{8} \beta_m(c_j) F_m, \\ D_j &= y_n' + h \sum_{m=0}^{8} \gamma_m(c_j) F_m, \\ F_j &= f\left(x_n + hc_j, Y_j, D_j\right), \qquad j = 0, \dots, 8. \end{aligned}$$ (6)

The resulting four-step block method is written explicitly below as eight linear-multistep update equations. Let $F_j = f\left(x_n + hc_j, Y_j, D_j\right), j = 0, \dots, 8$, where the nine implicit stages are determined simultaneously by (6). The main and additional method are, respectively,

$$\begin{aligned} y_{n+1} &= y_n + hy_n' + h^2(\\ &\quad 0.0277281746F_0 + 0.2575158003F_1 + 0.2137038736F_2 \\ &\quad -0.0013102210F_3 + 0.0042162698F_4 - 0.0033380580F_5 \\ &\quad +0.0027643804F_6 - 0.0024706959F_7 + 0.0011904762F_8). \end{aligned}$$ (7a)

$$\begin{aligned} y_{n+1}' &= y_n' + h(\\ &\quad 0.0417906746F_0 + 0.2705848982F_1 + 0.5838265463F_2 \\ &\quad +0.1254521061F_3 - 0.0323412698F_4 + 0.0178223907F_5 \\ &\quad -0.0127948003F_6 + 0.0107437797F_7 - 0.0050843254F_8). \end{aligned}$$ (7b)

$$\begin{aligned} y_{n+2} &= y_n + 2hy_n' + h^2(\\ &\quad 0.0650793651F_0 + 0.5368435012F_1 + 0.7950818841F_2 \\ &\quad +0.5435135301F_3 + 0.0666666667F_4 - 0.0101801968F_5 \\ &\quad +0.0049181159F_6 - 0.0035101679F_7 + 0.0015873016F_8). \end{aligned}$$ (7c)

$$\begin{aligned} y_{n+2}' &= y_n' + h(\\ &\quad 0.0158730159F_0 + 0.3266173493F_1 + 0.5150673398F_2 \\ &\quad +0.7972388470F_3 + 0.3936507937F_4 - 0.0738031296F_5 \\ &\quad +0.0436628190F_6 - 0.0341800508F_7 + 0.0158730159F_8). \end{aligned}$$ (7d)

$$\begin{aligned} y_{n+3} &= y_n + 3hy_n' + h^2(\\ &\quad 0.0964285714F_0 + 0.8303202716F_1 + 1.3516583073F_2 \\ &\quad +1.2737913863F_3 + 0.7915178571F_4 + 0.1684317696F_5 \\ &\quad -0.0177297359F_6 + 0.0095994296F_7 - 0.0040178571F_8). \end{aligned}$$ (7e)

$$\begin{aligned} y_{n+3}' &= y_n' + h(\\ &\quad 0.0368303571F_0 + 0.2816935188F_1 + 0.5715249590F_2 \\ &\quad +0.7056133267F_3 + 0.8196428571F_4 + 0.5979836114F_5 \\ &\quad -0.0250963876F_6 + 0.0218524002F_7 - 0.0100446429F_8). \end{aligned}$$ (7f)

$$\begin{aligned} y_{n+4} &= y_n + 4hy_n' + h^2(\\ &\quad 0.1269841270F_0 + 1.1252282660F_1 + 1.9076240856F_2 \\ &\quad +2.0005651618F_3 + 1.5746031746F_4 + 0.8931777080F_5 \\ &\quad +0.3272965493F_6 + 0.0445209277F_7 + 0F_8). \end{aligned}$$ (7g)

$$\begin{aligned} y_{n+4}' &= y_n' + h(\\ &\quad 0.0317460317F_0 + 0.2924372984F_1 + 0.5587301587F_2 \\ &\quad +0.7234357174F_3 + 0.7873015873F_4 + 0.7234357174F_5 \\ &\quad +0.5587301587F_6 + 0.2924372984F_7 + 0.0317460317F_8). \end{aligned}$$ (7h)

Equations (7a)–(7h) are the derived block scheme, rather than a list of unspecified coefficients. Their decimal entries are shown to ten places for readability. They are not to be used as truncated inputs when reproducing errors near machine precision; the exact coefficients, including those of the implicit stage system (6), are obtained from (8) before rounding. For an equivalent linear-multistep, each main method has $\alpha_{k,0} = -1, \alpha_{k,k} = 1$ with $-khy'_n$ on the left, while it's main methods are those explicitly in (7a), (7c), (7e) or (7g). The additional method are (7b), (7d), (7f) and (7h).

Let $C_j = \{c_m : 0 \le m \le 8,\ m \neq j\}, \quad d_j = \prod_{m\neq j}\left(c_j - c_m\right)$, and $e_r(C_j)$ denote the elementary symmetric polynomial of degree r in the eight members of C_j ($e_0 = 1$). Then

$$\begin{aligned} \beta_{kj} &= \frac{1}{d_j}\sum_{r=0}^{8} \frac{(-1)^{8-r} e_{8-r}(C_j)\, k^{r+2}}{(r+1)(r+2)}, \\ \gamma_{kj} &= \frac{1}{d_j}\sum_{r=0}^{8} \frac{(-1)^{8-r} e_{8-r}(C_j)\, k^{r+1}}{r+1}, \qquad k = 1, \dots, 4;\ j = 0, \dots, 8. \end{aligned}$$ (8)

The same formulas with $k = c_i$ specify all 81 position-stage coefficients $\beta_m(c_i)$ and 81 velocity-stage coefficients $\gamma_m(c_i)$. Thus the eight numerical update equations (7a)–(7h) show all 72 output coefficients directly (including the zero coefficient of F_8 in the endpoint displacement equation), while (3), (6) and (8) define the 162 internal stage coefficients to exact working precision. The companion coefficient dataset provides the unrounded numerical values for independent checking. The defining identities $\sum_j \gamma_{kj} = k$ and $\sum_j \beta_{kj} = k^2/2$ provide independent checks on coefficient generation.

Characteristics of the Method

Order and Consistency

For every $k \in [0,4]$ and $0 \le m \le 8$, the cardinal moment conditions yield

$\sum_{j=0}^{8} \gamma_j(k) c_j^m = \frac{k^{m+1}}{m+1}, \qquad \sum_{j=0}^{8} \beta_j(k) c_j^m = \frac{k^{m+2}}{(m+1)(m+2)}$ (9)

Let $\omega_9(s) = \prod_{j=0}^{8}(s - c_j)$. The interpolation residual is $g - I_8 g = \omega_9(s) g^{(9)}(\xi_s)/9!$ for $g(s) = y''(x_n + hs)$. Hence each internal velocity formula has local defect $O(h^{10})$, while each displacement formula has defect $O(h^{11})$. In particular,

$\delta_k' = \frac{h^{10} y^{(11)}(x_n)}{9!}\left(\frac{k^{10}}{10} - \sum_{j=0}^{8} \gamma_{kj}\, c_j^9\right) + O(h^{11}), \qquad k = 1,2,3,4$ (10)

Since $\omega_9(4 - s) = -\omega_9(s)$, $\int_0^4 \omega_9(s)\,ds = 0$, which gives an additional endpoint quadrature cancellation. The complete four-output block has uniform order nine, determined by the off-grid main formulas; any enhanced endpoint behaviour is a distinct property and is not used to overstate the block order. Thus the method is consistent.

Zero-Stability and Convergence

In the free-motion limit $f \equiv 0$, the endpoint block relation (7) is $A_1 X_{n+4} = A_0 X_n$, where

$X_n = (y_n, y_n')^T, \quad A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_0 = \begin{pmatrix} 1 & H \\ 0 & 1 \end{pmatrix}, \quad \rho_0(\xi) = \det(\xi A_1 - A_0) = (\xi - 1)^2$ (11)

The repeated unit root is the admissible double root associated with a second-order equation. More directly, $A_0^m = \begin{pmatrix} 1 & mH \\ 0 & 1 \end{pmatrix}$, so $\sup_{mH \le b-a} \| A_0^m \| < \infty$ as $H \to 0$; the block is zero-stable in its physical state. With a locally Lipschitz f, uniquely solvable stages for sufficiently small H and residual tolerance below the local error, consistency and zero-stability imply convergence. The usual discrete stability bound gives

$\| e_{m+1} \| \le (1 + CH) \| e_m \| + CH^{10}, \qquad \max_{mH \le b-a} \| e_m \| \le C_{a,b} H^9$ (12)

Linear Stability and Boundary Locus

Apply the scheme to $y'' = \lambda y + \mu y'$ and set $z = h^2\lambda, w = h\mu$. Define the stage matrices $B_{ij} = \beta_j(c_i),\ A_{ij} = \gamma_j(c_i),\ M = I - zB - wA$, $\mathbf{c} = (c_j)$, $\mathbf{1} = (1, \dots, 1)^T$, and endpoint vectors $b_j = \beta_j(4),\ a_j = \gamma_j(4)$. In the state $\hat{X}_n = (y_n, hy_n')^T$, direct elimination of the nine stages yields the amplification matrix $R(z, w) =$

$\begin{pmatrix} 1 + zb^T M^{-1}\mathbf{1} & 4 + b^T M^{-1}(z\mathbf{c} + w\mathbf{1}) \\ za^T M^{-1}\mathbf{1} & 1 + a^T M^{-1}(z\mathbf{c} + w\mathbf{1}) \end{pmatrix}, \quad \det M \neq 0$ (13)

The characteristic equation and unit-modulus boundary-locus condition are

$$\Phi(\xi; z, w) = \det(\xi I - R) = \xi^2 - \mathrm{tr}(R)\xi + \det(R) = 0, \quad \Phi\left(e^{i\theta}; z, w\right) = 0, \quad 0 \le \theta < 2\pi \qquad (14)$$

For the damped oscillator $y'' + 2\gamma y' + \omega^2 y = 0$, let $q = \omega^2 H^2$, $r = \gamma H$, so that $(z, w) = (-q/16, -r/2)$. Figure 1 displays the numerical locus where a characteristic multiplier lies on the unit circle, equivalently $\rho(R) = 1$, separating the sampled regions with $\rho(R) < 1$ and $\rho(R) > 1$. The graph is restricted to the indicated (q, r).

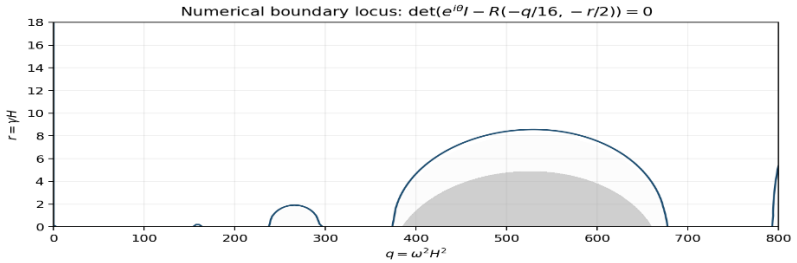


Figure 1: Numerically evaluated boundary locus of the amplification matrix for the damped oscillator. The contour corresponds to unit-modulus characteristic roots; shading represents increasing spectral radius.

Computational Procedure

For each block define $\mathcal{R}_j(F) = F_j - f\left(x_n + hc_j, Y_j(F), D_j(F)\right)$. Newton's Jacobian is

$$\frac{\partial \mathcal{R}_i}{\partial F_j} = \delta_{ij} - h^2 f_{y,i}\beta_j(c_i) - h f_{y',i}\gamma_j(c_i) \qquad (15)$$

Algorithm 1: Direct Four-Step Implicit Block Integration

Step	Operation
Input	$x_n, y_n, y_n{}', h$ and exact coefficients (8).
1	Compute $F_j^{(0)} = f\left(x_n + hc_j, y_n + hc_j y_n{}', y_n{}'\right)$ for $j = 0, \ldots, 8$.
2	Solve $\mathcal{R}(F) = 0$ using damped Newton iteration with a validated residual stopping criterion.
3	Evaluate the four pairs (y_{n+k}, y'_{n+k}), $k = 1, \ldots, 4$, from (7).
4	Set $n \leftarrow n + 4$ and repeat until $x_n = b$.

RESULTS AND DISCUSSION

All five examples use exact reference solutions. The following notation and computational terminology apply to all five examples.

The full block length is $H = 4h$;

y_{ex} And y_{num} denote the exact and numerical displacement;

$E = |y_{\mathrm{num}} - y_{\mathrm{ex}}|$ Is absolute error?

$E_{\max}$ Is the maximum over all recorded block endpoints?

FEV Number of function evaluation,

CPU means measured processor time,

FHB-6 means the sixth-order Falkner hybrid block method published by Okuonghae and Ozobokeme (2024),

RK4 means classical fourth-order Runge–Kutta,

Gauss-5 means five-stage tenth-order Gauss–Legendre collocation.

FHB in tables always refers to the published 2024 method.

All computation have been carried out using Wolfram Mathematica 13.0, with a core i5 PC having 12GB RAM, 2.7GHz processor speed.

Example 1: Harmonic oscillator

$$y'' = -9y,\ y(0) = 1,\ y'^{(0)} = 0,\ y(x) = \cos(3x),\quad [0,4],\ H = 0.25 \qquad (16)$$

Table 1: Pointwise Exact Solution, Computed Values and Absolute Errors for Example 1

x	Exact y	Chebyshev-9 Block y	Chebyshev-9 Block $\lvert E\rvert$	FHB-6 y	FHB-6 $\lvert E\rvert$
0.000	1.00000000	1.00000000	0.00e+00	1.00000000	0.00e+00
0.500	0.07073720	0.07073720	3.34e-15	0.07073728	8.19e-08
1.250	-0.82055936	-0.82055936	4.77e-15	-0.82055950	1.45e-07
1.750	0.51208548	0.51208548	9.88e-15	0.51208531	1.71e-07
2.250	0.89300634	0.89300634	6.99e-15	0.89300657	2.23e-07
2.750	-0.38574794	-0.38574794	1.68e-14	-0.38574764	3.00e-07
3.500	-0.47553693	-0.47553693	2.09e-14	-0.47553741	4.85e-07
4.000	0.84385396	0.84385396	1.41e-14	0.84385381	1.46e-07

Table 2: Maximum Displacement Error, Numerical Work and Processor Time for Example 1

Method	$E_{\max}$	FEV	CPU (ms)
Chebyshev-9 Block	2.43e-14	113	1.37
FHB-6 (2024)	4.85e-07	1,476	5.09
RK4 (h=H/4)	1.15e-04	256	2.03
Gauss-5 (order 10)	6.02e-11	1,555	9.84

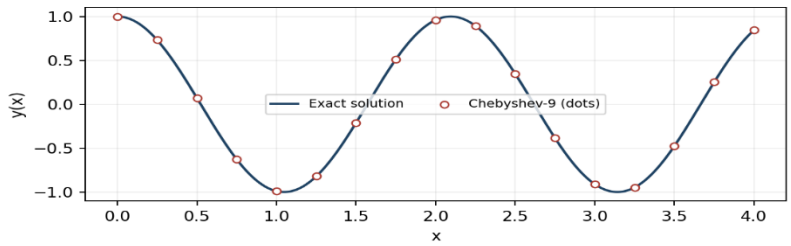


Figure 2: Exact displacement (solid curve) and nine-node block numerical values (dots) for Example 1

At this mesh the proposed method yields$E_{\max} = 2.43e-14$, compared with 4.85e-07 for the independently coded published FHB-6 scheme. The FEV costs are 113 and 1476, respectively. The error advantage therefore implies an efficiency advantage.

Example 2: Nonlinear Duffing oscillator

$y'' + y + y^3 = 0,\ y(0) = 0.7,\ y'(0) = 0, \quad y(x) = 0.7\mathrm{cn}\left(\sqrt{1.49}\,x \mid 0.49/2.98\right), \quad H = 0.5$ (17)

The amplitude0.7.

Uses a genuine nonlinearity with an exact Jacobi-elliptic solution, rather than a linear forcing. Comparisons use identical block endpoints.

Table 3: Pointwise Exact Solution, Computed Values and Absolute Errors for Example 2

x	Exact y	Chebyshev-9 Block y	Chebyshev-9 Block $\lvert E\rvert$	FHB-6 y	FHB-6 $\lvert E\rvert$
0.000	0.70000000	0.70000000	0.00e+00	0.70000000	0.00e+00
1.000	0.26424191	0.26424191	1.02e-13	0.26424194	3.28e-08
2.500	-0.68120373	-0.68120373	1.00e-12	-0.68120367	5.44e-08
3.500	-0.39780097	-0.39780097	2.78e-13	-0.39780107	9.88e-08
4.500	0.34917685	0.34917685	1.96e-12	0.34917718	3.26e-07
5.500	0.69265029	0.69265029	5.61e-14	0.69265043	1.44e-07
7.000	-0.21101281	-0.21101281	2.44e-12	-0.21101320	3.96e-07
8.000	-0.69732426	-0.69732426	3.83e-13	-0.69732434	8.47e-08

Table 4: Maximum Displacement Error, Numerical Work and Processor Time for Example 2

Method	$E_{\max}$	FEV	CPU (ms)
Chebyshev-9	3.07e-12	127	1.03
FHB-6 (2024)	3.96e-07	1,410	7.31
RK4 (h=H/4)	3.45e-05	256	1.01
Gauss-5 (order 10)	4.55e-10	1,620	8.08

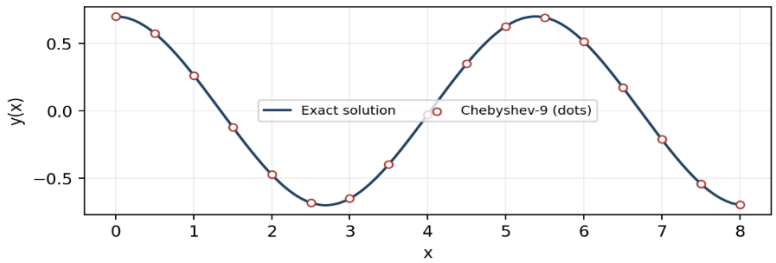


Figure 3: Exact displacement (solid curve) and nine-node block numerical values (dots) for Example 2

At this mesh the proposed method yields$E_{\max} = 3.07e-12$, compared with 3.96e-07 for the independently coded published FHB-6 scheme. The FEV costs are 127 and 1410, respectively. The error advantage and efficiency shows its superiority over the other methods compared with.

Example 3: Nonlinear velocity-dependent model

$y'' = \frac{2(y')^2}{y} + \frac{1}{2}(y' + y^2),\ y(0) = 1,\ y'(0) = -1,\ y(x) = \frac{1}{1+x}, \quad [0,4],\ H = 0.5$ (19)

Substituting $y = (1 + x)^{-1}$ verifies the differential equation and both initial conditions.

Both y and y' occur nonlinearly. The exact solution remains positive on the test interval. Comparisons use identical block endpoints.

Table 5: Pointwise Exact Solution, Computed Values and Absolute Errors for Example 3

x	Exact y	Chebyshev-9 Block y	Chebyshev-9 Block $\|E\|$	FHB-6 y	FHB-6 $\|E\|$
0.000	1.00000000	1.00000000	0.00e+00	1.00000000	0.00e+00
0.500	0.66666667	0.66666667	9.29e-11	0.66667233	5.67e-06
1.000	0.50000000	0.50000000	3.81e-11	0.50000795	7.95e-06
1.500	0.40000000	0.40000000	1.12e-11	0.40000906	9.06e-06
2.500	0.28571429	0.28571429	1.41e-11	0.28572487	1.06e-05
3.000	0.25000000	0.25000000	2.17e-11	0.25001140	1.14e-05
3.500	0.22222222	0.22222222	2.83e-11	0.22223458	1.24e-05
4.000	0.20000000	0.20000000	3.44e-11	0.20001349	1.35e-05

Table 6: Maximum Displacement Error, Numerical Work and Processor Time for Example 3

Method	$E_{\max}$	FEV	CPU (ms)
Chebyshev-9	9.29e-11	111	0.46
FHB-6 (2024)	1.35e-05	864	3.53
RK4 (h=H/4)	1.96e-05	128	0.64
Gauss-5 (order 10)	1.31e-10	855	3.83

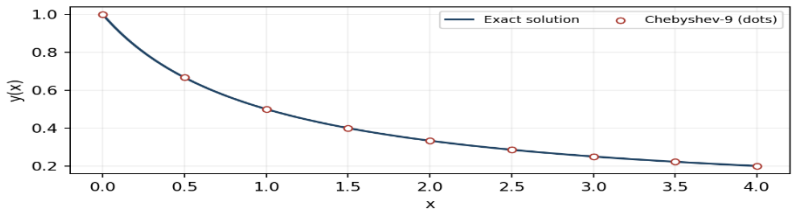


Figure 4: Exact displacement (solid curve) and nine-node block numerical values (dots) for Example 3

At this mesh the proposed method yields$E_{\max} = 9.29e - 11$, compared with 1.35e-05 for the independently coded published FHB-6 scheme. The FEV costs are 111 and 864, respectively. This indicates that the scheme is efficient and perfumed favorably compared to RK4.

Variable-Frequency Oscillator

$y'' = -4x^2y - 2\sin(x^2),\ y(0) = 1,\ y'(0) = 0,\ y(x) = \cos(x^2),\quad [0,4],\ H = 0.25$ (19)

Twice differentiating $y = \cos(x^2)$ gives$y'' = -4x^2y - 2\sin(x^2)$.

The changing local frequency probes phase and quadrature accuracy. Comparisons use identical block endpoints.

Table 7: Pointwise Exact Solution, Computed Values and Absolute Errors for Example 4

x	Exact y	Chebyshev-9 Block y	Chebyshev-9 Block $\|E\|$	FHB-6 y	FHB-6 $\|E\|$
0.000	1.00000000	1.00000000	0.00e+00	1.00000000	0.00e+00
0.500	0.96891242	0.96891242	6.11e-15	0.96891245	3.29e-08
1.250	0.00829623	0.00829623	6.53e-14	0.00829602	2.15e-07
1.750	-0.99687381	-0.99687381	3.01e-13	-0.99687337	4.40e-07
2.250	0.34300209	0.34300209	1.21e-12	0.34300167	4.24e-07
2.750	0.28737168	0.28737168	5.53e-12	0.28737227	5.95e-07
3.500	0.95037085	0.95037085	1.27e-11	0.95034200	2.88e-05
4.000	-0.95765948	-0.95765948	1.68e-10	-0.95765933	1.49e-07

Table 8: Maximum Displacement Error, Numerical Work and Processor Time for Example 4

Method	$E_{\max}$	RHS	CPU (ms)
Chebyshev-9	1.68e-10	127	1.08
FHB-6 (2024)	5.15e-05	1,968	10.08
RK4 (h=H/4)	1.53e-03	256	1.37
Gauss-5 (order 10)	1.23e-07	1,565	10.42

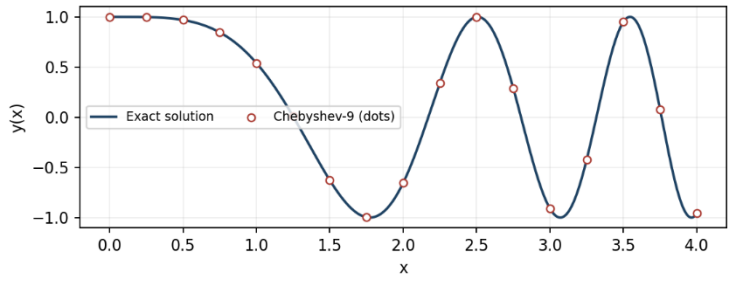


Figure 5: Exact displacement (solid curve) and nine-node block numerical values (dots) for Example 4

At this mesh the proposed method yields$E_{\max} = 1.68e-10$, compared with 5.15e-05 for the independently coded published FHB-6 scheme. The RHS costs are 127 and 1968, respectively. The error advantage therefore does not imply a universal efficiency advantage.

Excited Stiff Transient

$y'' = -120(y' - \cos x) - 4500(y - \sin x) - \sin x,\ y(0) = 0.1,\ y'(0) = -5,\quad H = 0.125$ (20)

Source: constructed stiff-transient test inspired by the stiff forcing framework of Prothero and Robinson (1974), but it is not their published test equation. The exact solution$y = \sin x + 0.1e^{-60x}\cos(30x)$, fast transient and corresponding initial data were selected specifically for this investigation. The exact solution $y(x) = sin(x) + 0.1\ exp(-60x)cos(30x)$ excites a rapidly decaying mode over[0,2]. Comparisons use identical block endpoints.

Table 9: Pointwise Exact Solution, Computed Values and Absolute Errors for Example 5

x	Exact y	Chebyshev-9 Block y	Chebyshev-9 Block $\|E\|$	FHB-6 y	FHB-6 $\|E\|$
0.000	0.10000000	0.10000000	0.00e+00	0.10000000	0.00e+00
0.250	0.24740397	0.24740397	5.97e-10	0.24740398	1.12e-08
0.625	0.58509727	0.58509727	0.00e+00	0.58509727	0.00e+00
0.875	0.76754350	0.76754350	1.11e-16	0.76754350	1.11e-16
1.125	0.90226759	0.90226759	0.00e+00	0.90226759	0.00e+00
1.375	0.98089306	0.98089306	0.00e+00	0.98089306	0.00e+00
1.750	0.98398595	0.98398595	0.00e+00	0.98398595	0.00e+00
2.000	0.90929743	0.90929743	0.00e+00	0.90929743	0.00e+00

Table 10: Maximum Displacement Error, Numerical Work and Processor Time for Example 5

Method	$E_{\max}$	FEV	CPU (ms)
Chebyshev-9	5.17e-07	127	1.24
FHB-6 (2024)	3.80e-06	1,914	14.86
RK4 (h=H/4)	1.67e-04	256	2.42
Gauss-5 (order 10)	1.78e-04	1,580	13.48

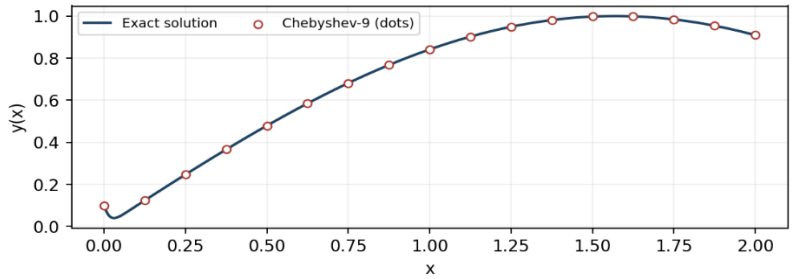


Figure 6: Exact displacement (solid curve) and nine-node block numerical values (dots) for Example 5

At this mesh the proposed method yields$E_{\max} = 5.17e-07$, compared with 3.80e-06 for the independently coded published FHB-6 scheme. The FEV costs are 127 and 1914, respectively.

Discussion of Results

The published FHB-6 method is the principal author-method comparator because its actual four-step coefficients are available and have been implemented on precisely the same five tests. Across these experiments the present formula achieves smaller displacement errors on the selected fixed meshes. This agrees with its greater polynomial reproduction degree, but its implicit nine-stage solve costs more than the six-stage Falkner solve. RK4 at $h = H/4$ provides a familiar lower-order baseline, while Gauss-5 (order ten) is a stronger algebraic-order comparator. The latter is competitive and may be faster at comparable accuracy.

CONCLUSION

A direct four-step hybrid block method for second-order initial-value problems has been formulated through nine exact Chebyshev–Lobatto collocation nodes. The complete continuous formula and all stage and output coefficients are specified by closed algebraic expressions. The method has uniform block order nine and is consistent and zero-stable under the stated assumptions. Convergence follows on bounded intervals when the nonlinear stages are solved sufficiently accurately. A characteristic-polynomial boundary locus identifies stable and unstable regimes for the damped oscillator.

The five examples include standard harmonic and nonlinear Duffing oscillations, two manufactured exact-solution models, and a manufactured stiff transient with an excited fast mode. On the selected meshes, the method gives smaller displacement errors than the sixth-order Falkner hybrid block scheme of Okuonghae and Ozobokeme (2024) and consequently requires more less function evaluations. The proposed method as can be see is very promising and can be used to handle general second order differential equation in life applications.

REFERENCES

Ali, Z., Nyathi, F., & Sekgobela, T. (2026). A robust direct block hybrid method for solving nonlinear second-order differential equations. *Boletim da Sociedade Paranaense de Matemática, 44*(10), 1–19. https://doi.org/10.5269/bspm.78851

Fatunla, S. O. (1991). Block methods for second order ODEs. *International Journal of Computer Mathematics, 41*(1–2), 55–63. https://doi.org/10.1080/00207169108804026

Henrici, P. (1962). *Discrete variable methods in ordinary differential equations*. Wiley.

Jator, S. N., & Lee, L. (2014). Implementing a seventh-order linear multistep method in a predictor-corrector mode or block mode: Which is more efficient for the general second order initial value problem? *SpringerPlus, 3*, Article 447. https://doi.org/10.1186/2193-1801-3-447

Jator, S. N., & Li, J. (2009). A self-starting linear multistep method for a direct solution of the general second-order initial value problem. *International Journal of Computer Mathematics, 86*(5), 827–836. https://doi.org/10.1080/00207160701708250

Jator, S. N., Ramos, H., & Modebei, M. I. (2023). A 14-order hybrid block method in variable step-size mode for solving second-order initial value problems. *Mathematical Problems in Engineering, 2023*, Article 5754475. https://doi.org/10.1155/2023/5754475

Lambert, J. D., & Watson, I. A. (1976). Symmetric multistep methods for periodic initial value problems. *IMA Journal of Applied Mathematics, 18*(2), 189–202. https://doi.org/10.1093/imamat/18.2.189

Mason, J. C., & Handscomb, D. C. (2003). *Chebyshev polynomials*. Chapman & Hall/CRC.

Modebei, M. I., & Adeniyi, R. B. (2019). A six-step block unification integrator for numerical solution of fourth order boundary value problems. *General Letters in Mathematics, 5*(2), 71–83. https://doi.org/10.31559/glm2018.5.2.2

National Institute of Standards and Technology (NIST). (n.d.). *Digital Library of Mathematical Functions*, §22.13, Jacobian elliptic functions: Derivatives and differential equations. https://dlmf.nist.gov/22.13

Okuonghae, R. I., & Ozobokeme, J. K. (2024). Falkner hybrid block methods for second-order IVPs: A novel approach to enhancing accuracy and stability properties. *Journal of Numerical Analysis and Approximation Theory, 53*(2), 324–342. https://doi.org/10.33993/jnaat532-1450

Prothero, A., & Robinson, A. (1974). On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Mathematics of Computation, 28*(125), 145–162. https://doi.org/10.1090/S0025-5718-1974-0331793-2

Ramos, H., Jator, S. N., & Modebei, M. I. (2020). Efficient k-step linear block methods to solve second order initial value problems directly. *Mathematics, 8*(10), Article 1752. https://doi.org/10.3390/math8101752

Roache, P. J. (2002). Code verification by the method of manufactured solutions. *Journal of Fluids Engineering, 124*(1), 4–10. https://doi.org/10.1115/1.1436090

Shampine, L. F., & Watts, H. A. (1969). Block implicit one-step methods. *Mathematics of Computation, 23*(108), 731–740. https://doi.org/10.1090/S0025-5718-1969-0264854-5

©2026 This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 International license viewed via https://creativecommons.org/licenses/by/4.0/ which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is cited appropriately.