Available online www.worldscientificnews.com

World Scientific News

An International Scientific Journal

WSN 218 (2026) 127-138 EISSN 2392-2192

A Modified Hybrid Steffensen's Method Using Adaptive Weights for Solving Nonlinear Equations

Danny Darliansyah[1,*], Zhafir Al Haq Taqiyyuddin Ali Aqil Badruzzaman[1], Sri Purwani[2]

[1]Master Program in Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Jatinangor, Sumedang 45363, Indonesia

[2]Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Jatinangor, Sumedang 45363, Indonesia

E-mail address: danny25001@mail.unpad.ac.id

https://doi.org/10.65770/BNRC9106

ABSTRACT

This paper proposes a modified hybrid Steffensen method for solving nonlinear equations, in which the fixed convex combination weight of the standard hybrid scheme is replaced by an adaptive residual-based weight. At each step, the forward and backward Steffensen candidates are computed and combined through a weight that automatically concentrates on whichever candidate has the smaller residual. The convergence analysis establishes that the proposed scheme retains second-order convergence and, under the condition that the derivative of the function at the simple root lies strictly between zero and one, achieves a smaller asymptotic error constant than the equal-weight combination. Numerical experiments on twenty test functions of polynomial, trigonometric, exponential, logarithmic, and mixed type confirm that the adaptive variant requires fewer or equal iterations than the constant-weight method in every case considered, with the largest reduction observed where the fixed weight is locally inefficient. The number of function evaluations and the computational order of convergence are reported alongside the iteration counts to give a balanced view of the cost of the adaptive scheme.

Keywords: Nonlinear equations, Steffensen method, Adaptive weighting, Convex combination, Iterative methods.

(Received 11 June 2026; Accepted 10 July 2026; Date of Publication 31 August 2026)

1. INTRODUCTION

The Steffensen method has long been recognised as an effective derivative-free iterative technique for solving nonlinear equations. Its appeal lies in its simplicity and computational efficiency, particularly in situations where derivative evaluation is impractical or computationally expensive [8, 17, 18]. As research has progressed, various modifications have been proposed to improve convergence rates and broaden the range of applicability, including parametric variants, memory-based acceleration, and hybrid approaches [5, 9, 20, 21]. A common limitation of these methods is the use of fixed parameters or weights throughout the iteration, which constrains their adaptability to changing problem characteristics during the solution process.

Within the broader framework of iterative methods, adaptive weighting has emerged as a promising strategy for improving both convergence speed and robustness. Adaptive weighting refers to a mechanism that dynamically adjusts parameters at each iteration based on the evolving characteristics of the problem. Dynamic weight adjustment has been successfully applied to optimisation algorithms such as the Alternating Direction Method of Multipliers and to metaheuristic-based hybrid methods, where it has been shown to improve convergence speed and solution accuracy [14, 16]. In the numerical analysis literature, adaptive schemes have similarly been shown to reduce errors and improve computational efficiency by tailoring the algorithm to the local behaviour of the problem being solved [22].

Building on these ideas, the present work introduces an adaptive weight function w_n that changes at each iteration of a hybrid Steffensen scheme. Unlike conventional approaches that fix the weight in advance, this adaptive mechanism allows the method to respond dynamically to the local degree of nonlinearity and asymmetry between the forward and backward Steffensen candidates. The approach is inspired by dynamic weighting strategies in other computational fields, such as neural networks and genetic algorithms, where weights are iteratively updated to optimise performance [7, 23]. By incorporating this flexibility, the modified Steffensen method is expected to achieve improved convergence robustness and stability, particularly in cases where a fixed weight performs poorly because of a strong asymmetry between the forward and backward Steffensen candidates [18, 20].

The theoretical justification for this adaptive approach is supported by various studies on parametric and memory-based variants of the Steffensen method, which show that careful parameter tuning can reduce the asymptotic error constant and improve robustness across diverse problem configurations [4, 19, 21]. Numerical experiments in related work further confirm that adaptive schemes can reduce the computational burden without sacrificing accuracy, and in many cases improve it [9, 22]. The present paper develops a closed-form residual-based adaptive weight that requires no user-specified parameter, proves that the resulting method retains second-order convergence while strictly reducing the asymptotic error constant relative to the equal-weight hybrid under a mild condition on the derivative at the root, and demonstrates the improvement on a benchmark of twenty test functions through comparison of iteration counts, function evaluations, and computational order of convergence.

2. MATERIALS AND METHODS

2.1. Standard Definitions

Definition 1 (Nonlinear Equation) [13]. Let $f: D \subseteq \mathbb{R} \to \mathbb{R}$ be a real-valued function. An equation of the form $f(x) = 0$ is called a nonlinear equation if f is not expressible as a linear polynomial in x, that is, $f(x) \neq ax + b$ for any constants $a, b \in \mathbb{R}$ with $a \neq 0$. Such an equation may possess a unique solution, no solution at all, or multiple solutions, depending on the nature of f.

Definition 2 (Simple Root) [10, 11]. A point $x^* \in D$ is called a simple root of the equation $f(x) = 0$ if $f(x^*) = 0$ and $f'(x^*) \neq 0$. The condition $f'(x^*) \neq 0$ ensures that f cuts the x-axis transversally at x^* and does not merely become tangent to it.

Definition 3 (Iterative Method) [2, 13]. An iterative method for solving $f(x) = 0$ is a procedure that generates a sequence $\{x_n\}$ from an initial approximation x_0 via the recursive relation

$$x_{n+1} = g(x_n),$$

such that $x_n \to x^*$ as $n \to \infty$, whenever x_0 is chosen sufficiently close to the exact root x^*. The function g is referred to as the iteration function of the method.

Definition 4 (Order of Convergence) [1, 6, 9]. Suppose $\{x_n\}$ is a sequence converging to x^*. The error at the n-th iteration is defined as $e_n = x_n - x^*$. A method is said to converge with order $p \geq 1$ if there exists a nonzero constant C, called the asymptotic error constant, such that

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^p} = C.$$

When $p = 1$ the method exhibits linear convergence, when $p = 2$ quadratic convergence, and so on. A higher value of p implies more rapid error decrease with each successive iteration.

Definition 5 (Error Equation) [15]. If $e_n = x_n - \alpha$ denotes the error at the n-th iteration, where α is the exact root, then the equation

$$e_{n+1} = C\, e_n^p + O\big(e_n^{p+1}\big)$$

is called the error equation. The value p represents the order of convergence and C is the asymptotic error constant.

2.2. Newton-Raphson Method

The Newton-Raphson method solves a nonlinear equation from a single starting point by exploiting the slope (gradient) of f [3, 12]. Let x_n be the current approximation. The first-order Taylor expansion of f about x_n, evaluated at the root α, gives

$$f(\alpha) = f(x_n) + f'(x_n)(\alpha - x_n) + \frac{f''(\xi)}{2}(\alpha - x_n)^2$$

for some ξ between x_n and α. Neglecting the second-order term and using $f(\alpha) = 0$ yields the Newton-Raphson iteration formula

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \qquad f'(x_n) \neq 0. \tag{1}$$

The convergence order of the Newton-Raphson method is well known. Suppose α is a simple root of $f(x) = 0$ and $f \in C^2$ in a neighbourhood of α. Then the method converges quadratically with $p = 2$ and asymptotic error constant

$$C_{\mathrm{NR}} = \frac{|f''(\alpha)|}{2|f'(\alpha)|}.$$

Proof of convergence order. From the Taylor expansions about α,

$$f(x_n) = f'(\alpha)\, e_n + \frac{1}{2} f''(\alpha)\, e_n^2 + O(e_n^3),$$

$$f'(x_n) = f'(\alpha) + f''(\alpha)\, e_n + O(e_n^2).$$

Substituting into (1) and using the approximation $(1+u)^{-1} \approx 1-u$ for small u gives

$$e_{n+1} = e_n - \frac{f'(\alpha)e_n + \frac{1}{2}f''(\alpha)e_n^2}{f'(\alpha)+f''(\alpha)e_n} = \frac{f''(\alpha)}{2f'(\alpha)}\,e_n^2 + O(e_n^3).$$

Hence $\lim_{n\to\infty}|e_{n+1}|/|e_n|^2 = |f''(\alpha)|/2|f'(\alpha)|$, confirming $p = 2$.

2.3. Forward Steffensen Method

To avoid evaluating $f'(x_n)$ in (1), the derivative is approximated by the forward divided difference with step length $h_n = f(x_n)$,

$$f'(x_n) \approx \frac{f(x_n+f(x_n)) - f(x_n)}{f(x_n)}.$$

Substituting into (1) yields the forward Steffensen candidate,

$$F_n = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n)) - f(x_n)}. \tag{2}$$

This formula approximates the root using information in the forward direction from x_n.

2.4. Backward Steffensen Method

Analogously, the derivative may be approximated by the backward divided difference with step length $h_n = f(x_n)$,

$$f'(x_n) \approx \frac{f(x_n) - f(x_n-f(x_n))}{f(x_n)},$$

which gives the backward Steffensen candidate,

$$B_n = x_n - \frac{f(x_n)^2}{f(x_n) - f(x_n-f(x_n))}. \tag{3}$$

2.5. Convex Combination

Given points $x_1, x_2, \ldots, x_k \in \mathbb{R}^n$, a point $z \in \mathbb{R}^n$ is called a convex combination of these points if there exist scalars $\lambda_1, \ldots, \lambda_k$ with $\lambda_i \geq 0$ and $\sum_{i=1}^{k} \lambda_i = 1$ such that $z = \sum_{i=1}^{k} \lambda_i\, x_i$. If $w \notin [0,1]$, the iterate $x_{n+1} = wF_n + (1-w)B_n$ may lie outside the interval $[F_n, B_n]$, which can lead to divergence or instability. The restriction $w \in [0,1]$ is therefore necessary for the combination to be geometrically meaningful. In the adaptive case, w is replaced by w_n satisfying $w_n \in [0,1]$ via a residual-based normalisation scheme, maintaining the convex structure at each step.

2.6. Hybrid Steffensen Method

Combining the two candidates (2) and (3) as a convex combination gives the hybrid Steffensen method [5],

$$x_{n+1} = w\,F_n + (1-w)\,B_n, \qquad 0 \leq w \leq 1, \tag{4}$$

which expands to

$$x_{n+1} = w\left(x_n - \frac{f(x_n)^2}{f(x_n+f(x_n)) - f(x_n)}\right) + (1-w)\left(x_n - \frac{f(x_n)^2}{f(x_n) - f(x_n-f(x_n))}\right).$$

The parameter w regulates the relative contribution of each candidate. When w is constant, the combination weights remain unchanged throughout the iteration.

3. RESULTS AND DISCUSSIONS

3.1. Construction of the Adaptive Weight

To increase the flexibility of (4), the constant parameter w is replaced by an iteration-dependent weight w_n,

$$x_{n+1} = w_n F_n + (1 - w_n) B_n, \qquad w_n \in [0,1]. \tag{5}$$

The construction of w_n is based on the residuals of the two candidates. Since the goal of iteration is to satisfy $f(x) = 0$, the quality of each candidate may be assessed through its residual,

$$r_n^{(F)} = |f(F_n)|, \qquad r_n^{(B)} = |f(B_n)|.$$

A smaller residual indicates a candidate closer to the root, so the weight assigned to a candidate should increase as its residual decreases. This motivates the quality measures

$$q_n^{(F)} = \frac{1}{r_n^{(F)}}, \qquad q_n^{(B)} = \frac{1}{r_n^{(B)}}.$$

Normalising so that $w_n + (1 - w_n) = 1$ (the convex constraint),

$$w_n = \frac{q_n^{(F)}}{q_n^{(F)} + q_n^{(B)}} = \frac{1/r_n^{(F)}}{1/r_n^{(F)} + 1/r_n^{(B)}}.$$

Multiplying numerator and denominator by $r_n^{(F)} r_n^{(B)}$ gives the closed-form residual-based weight,

$$w_n = \frac{|f(B_n)|}{|f(F_n)| + |f(B_n)|}. \tag{6}$$

When $|f(F_n)|$ is large relative to $|f(B_n)|$, the forward candidate lies further from the root and w_n decreases, concentrating weight on the more accurate backward candidate, and vice versa.

3.2. Convergence Analysis

Theorem 6. Let α be a simple root of $f(x) = 0$, with f three times continuously differentiable in a neighbourhood of α. Define $e_n = x_n - \alpha$, $c_1 = f'(\alpha)$, and $c_2 = f''(\alpha)/2f'(\alpha)$. Then the following statements hold.

(I) The forward candidate satisfies $F_n - \alpha = c_2\big(1 + f'(\alpha)\big)e_n^2 + O(e_n^3)$.

(II) The backward candidate satisfies $B_n - \alpha = c_2\big(1 - f'(\alpha)\big)e_n^2 + O(e_n^3)$.

(III) For a fixed weight $w \in [0,1]$, the hybrid Steffensen method converges to α with order $p = 2$ and asymptotic error constant

$$C = c_2[1 + (2w - 1)f'(\alpha)].$$

In particular, for $w = 1/2$ one has $C = c_2 = f''(\alpha)/2f'(\alpha)$, which coincides with the Newton-Raphson error constant.

Proof. Let $h_n = f(x_n)$. Since α is a simple root with $f'(\alpha) = c_1 \neq 0$, the Taylor expansion about α gives

$$f(x_n) = c_1 e_n + c_1 c_2 e_n^2 + O(e_n^3), \qquad h_n = O(e_n). \tag{7}$$

Proof of (I). Since $x_n + h_n - \alpha = (1 + c_1)e_n + c_1 c_2 e_n^2 + O(e_n^3)$, expanding $f(x_n + h_n)$ about α,

$$f(x_n + h_n) = c_1[(1 + c_1)e_n + c_1 c_2 e_n^2] + c_1 c_2[(1 + c_1)e_n]^2 + O(e_n^3)$$

$$= c_1(1+c_1)\,e_n + c_1c_2[c_1 + (1+c_1)^2]e_n^2 + O(e_n^3)$$

$$= c_1(1+c_1)\,e_n + c_1c_2(1+3c_1+c_1^2)\,e_n^2 + O(e_n^3). \qquad (8)$$

Subtracting (7) from (8),

$$f(x_n+h_n) - f(x_n) = c_1^2 e_n[1 + c_2(3+c_1)e_n + O(e_n^2)]. \qquad (9)$$

From (7) one has $f(x_n)^2 = c_1^2 e_n^2[1 + 2c_2e_n + O(e_n^2)]$. Dividing and using $(1+u)^{-1} \approx 1-u$ for small u,

$$\frac{f(x_n)^2}{f(x_n+h_n)-f(x_n)} = e_n[1+2c_2e_n][1-c_2(3+c_1)e_n] + O(e_n^3)$$

$$= e_n - c_2(1+c_1)\,e_n^2 + O(e_n^3).$$

Therefore $F_n - \alpha = e_n - f(x_n)^2/[f(x_n+h_n) - f(x_n)] = c_2(1+c_1)e_n^2 + O(e_n^3)$, which proves part (I).

Proof of (II). Since $x_n - h_n - \alpha = (1-c_1)e_n - c_1c_2e_n^2 + O(e_n^3)$, expanding $f(x_n - h_n)$ about α,

$$f(x_n-h_n) = c_1(1-c_1)\,e_n + c_1c_2[(1-c_1)^2 - c_1]e_n^2 + O(e_n^3)$$

$$= c_1(1-c_1)\,e_n + c_1c_2(1-3c_1+c_1^2)\,e_n^2 + O(e_n^3).$$

Therefore $f(x_n) - f(x_n-h_n) = c_1^2e_n[1 + c_2(3-c_1)e_n + O(e_n^2)]$. Proceeding analogously to part (I),

$$\frac{f(x_n)^2}{f(x_n)-f(x_n-h_n)} = e_n - c_2(1-c_1)\,e_n^2 + O(e_n^3),$$

so $B_n - \alpha = c_2(1-c_1)e_n^2 + O(e_n^3)$, which proves part (II).

Proof of (III). For any fixed $w \in [0,1]$,

$$e_{n+1} = w(F_n-\alpha) + (1-w)(B_n-\alpha)$$

$$= w\,c_2(1+c_1)e_n^2 + (1-w)\,c_2(1-c_1)e_n^2 + O(e_n^3)$$

$$= c_2[(1-c_1) + 2w\,c_1]e_n^2 + O(e_n^3)$$

$$= c_2[1 + (2w-1)c_1]e_n^2 + O(e_n^3).$$

Since $C = c_2[1 + (2w-1)f'(\alpha)] \neq 0$ for generic values of $f'(\alpha)$ and w, the order $p = 2$ is confirmed. For $w = 1/2$ one has $C = c_2 = f''(\alpha)/2f'(\alpha)$, which is precisely the Newton-Raphson error constant.

Remark 7. Theorem 6 shows that the forward and backward Steffensen candidates carry different second-order error coefficients, namely $c_2\big(1+f'(\alpha)\big)$ for F_n and $c_2\big(1-f'(\alpha)\big)$ for B_n. The asymptotic error constant of the hybrid method therefore depends on both w and $f'(\alpha)$. The special choice $w = 1/2$ yields $C = c_2$ regardless of $f'(\alpha)$, because the leading-order contributions from the two candidates cancel symmetrically. The adaptive weight departs from $1/2$ whenever the residuals differ, concentrating weight on the more accurate candidate and adjusting the effective error constant at each step.

Corollary 8. Under the conditions of Theorem 6, suppose $0 < f'(\alpha) < 1$. The adaptive-weight hybrid Steffensen method converges with order $p = 2$. Moreover, as $n \to \infty$ the adaptive weight satisfies $w_n \to \big(1 - f'(\alpha)\big)/2$, yielding the asymptotic error constant

$$C_{\text{adaptive}} = c_2(1 - f'(\alpha)^2),$$

which satisfies $|C_{\text{adaptive}}| < |c_2| = |C_{w=1/2}|$. That is, the residual-based adaptive weight achieves a strictly smaller asymptotic error constant than the equal-weight combination, without increasing the convergence order.

Proof. From parts (I) and (II) of Theorem 6, near convergence one has

$$|f(F_n)| \approx |c_1||c_2|(1 + c_1)\, e_n^2, \qquad |f(B_n)| \approx |c_1||c_2|(1 - c_1)\, e_n^2.$$

For $0 < c_1 < 1$ both quantities are positive, so the adaptive weight (6) satisfies

$$w_n = \frac{|f(B_n)|}{|f(F_n)| + |f(B_n)|} \overset{n\to\infty}{\to} \frac{(1-c_1)}{(1+c_1)+(1-c_1)} = \frac{1-c_1}{2}.$$

Substituting $w_\infty = (1 - c_1)/2$ into the formula from Theorem 6 (III),

$$C_{\text{adaptive}} = c_2\left[1 + \left(2 \cdot \frac{1-c_1}{2} - 1\right) c_1\right] = c_2[1 - c_1^2] = c_2(1 - f'(\alpha)^2).$$

Since $0 < f'(\alpha) < 1$ implies $0 < 1 - f'(\alpha)^2 < 1$, one obtains $|C_{\text{adaptive}}| < |c_2|$.

3.3. Experimental Setup

The numerical experiments compare four methods, the forward Steffensen method (2), the backward Steffensen method (3), the constant-weight hybrid method (4) with $w = 0.5$, and the adaptive-weight variant (5) with weight (6). The adaptive variant starts with $w_0 = 0.50$; subsequent weights are updated from the candidate residuals, with a regularisation parameter $\varepsilon = 10^{-15}$ in the denominator of (6) to prevent division by near-zero values. Iteration terminates when $|f(x_n)| < \text{tol} = 10^{-12}$ or n reaches $N_{\max} = 1000$. The computational order of convergence (COC) is estimated from

$$\text{COC} = \frac{\ln|e_{n+1}/e_n|}{\ln|e_n/e_{n-1}|}, \tag{10}$$

evaluated at the last three iterates with nonzero error.

The twenty test functions and initial parameters are listed in Table 1. The collection spans polynomial, exponential, logarithmic, trigonometric, and mixed algebraic-transcendental equations, providing diverse local structures near the root. Initial guesses are chosen within the relevant domain, and all problems share $w_0 = 0.50$ so that the comparison isolates the effect of the adaptive update.

Table 1. Test Functions and Initial Parameters.

No.	**Nonlinear function**	x_0	w_0
1	$f_1(x) = x^3 - x - 2$	1.5	0.50
2	$f_2(x) = \cos x - x$	0.5	0.50
3	$f_3(x) = e^{-x} - x$	0.5	0.50
4	$f_4(x) = x^3 - 2x - 5$	2.0	0.50
5	$f_5(x) = x - \sin x - 0.5$	1.0	0.50
6	$f_6(x) = \ln(x + 1) + x - 2$	1.0	0.50

7	$f_7(x) = xe^x - 1$	0.5	0.50
8	$f_8(x) = x^5 - x - 1$	1.2	0.50
9	$f_9(x) = \tan x - x$	4.5	0.50
10	$f_{10}(x) = x^3 + 4x^2 - 10$	1.5	0.50
11	$f_{11}(x) = x - e^{-x}$	0.0	0.50
12	$f_{12}(x) = x^3 - 1$	0.5	0.50
13	$f_{13}(x) = \sin x - x/2$	1.5	0.50
14	$f_{14}(x) = x^2 - e^{-x} - 3x + 2$	0.0	0.50
15	$f_{15}(x) = \cos x - xe^x$	1.0	0.50
16	$f_{16}(x) = x^4 - x - 1$	1.5	0.50
17	$f_{17}(x) = xe^{-x} - 0.1$	0.5	0.50
18	$f_{18}(x) = x^3 + x - 1$	1.0	0.50
19	$f_{19}(x) = e^x - 3x^2$	1.0	0.50
20	$f_{20}(x) = x^2 - (1 - x)^5$	0.5	0.50

3.4. Numerical Results

Table 2 reports the numerical results. All four methods use the same initial guess, stopping tolerance, and maximum iteration limit, providing a direct comparison of iteration counts (N), and number of function evaluations (NFE).

Table 2. Numerical Results.

No.	Fn	α	x_0	Forward			Backward			Hybrid			Adaptive		
				N	NFE	COC	N	NFE	COC	N	NFE	COC	N	NFE	COC
1	f_1	1.521379706805	1.5	4	9	2.00	4	9	2.00	3	10	2.00	3	14	1.31
2	f_2	0.739085133215	0.5	4	9	2.00	5	11	2.00	4	13	2.00	3	14	1.28
3	f_3	0.567143290410	0.5	3	7	2.00	4	9	2.00	3	10	2.00	2	9	5.97
4	f_4	2.094551481542	2.0	7	15	2.00	6	13	2.00	4	13	2.00	3	14	3.86
5	f_5	1.497300389096	1.0	6	13	2.00	4	9	2.00	5	16	2.00	4	19	2.06
6	f_6	1.207940031569	1.0	4	9	2.00	3	7	2.00	3	10	2.00	2	9	6.13
7	f_7	0.567143290410	0.5	5	11	2.00	4	9	2.00	4	13	2.00	3	14	1.34
8	f_8	1.167303978261	1.2	6	13	2.00	6	13	2.00	4	13	2.00	3	14	1.80
9	f_9	4.493409457909	4.5	6	13	2.00	6	13	2.00	3	10	2.00	3	14	1.28
10	f_{10}	1.365230013414	1.5	7	15	2.00	12	25	2.00	38	115	2.00	5	24	1.28
11	f_{11}	0.567143290410	0.0	5	11	2.00	4	9	2.00	4	13	2.00	3	14	1.76
12	f_{12}	1.000000000000	0.5	175	351	2.00	6	13	2.00	16	49	2.00	15	74	4.01
13	f_{13}	1.895494267034	1.5	4	9	2.00	6	13	2.00	5	16	2.00	5	24	1.88
14	f_{14}	0.608989103010	0.0	4	9	2.00	6	13	2.00	4	13	2.00	3	14	1.35
15	f_{15}	0.517757363682	1.0	5	11	2.00	10	21	2.00	8	25	—	4	19	4.00
16	f_{16}	1.220744084606	1.5	13	27	2.00	NC	—	—	7	22	—	5	24	—
17	f_{17}	0.111832559159	0.5	8	17	2.00	4	9	2.00	6	19	2.00	4	19	2.00
18	f_{18}	0.682327803828	1.0	7	15	2.00	6	13	2.00	4	13	2.00	3	14	1.35
19	f_{19}	0.910007572489	1.0	4	9	2.00	5	11	2.00	4	13	2.00	3	14	1.30
20	f_{20}	0.345954815848	0.5	4	9	2.00	4	9	2.00	4	13	2.00	3	14	1.37

Three exceptions to convergence toward the tabulated root are recorded. The backward Steffensen method fails to converge on f_{16} within the iteration limit. On f_{15} the constant-weight hybrid method, and on f_{16} both the constant-weight hybrid and the adaptive variant, satisfy the stopping criterion at a different real root of the same equation rather than at the root listed in Table 2. For the remaining entries all methods reach the tabulated root. The adaptive method matches or improves the iteration count of the constant-weight hybrid in every test case considered. Out of the twenty functions, the adaptive variant uses strictly fewer iterations in seventeen cases and the same number in three cases (f_1, f_9, f_{13}). The mean iteration count for the adaptive method across the nineteen functions on which all four methods converge is 3.4, compared with 6.6 for the hybrid.

4. DISCUSSION

The most pronounced improvement occurs for $f_{10}(x) = x^3 + 4x^2 - 10$, where the constant-weight hybrid method requires 38 iterations while the adaptive variant converges in only 5. The mechanism is explained by Remark 7 and Corollary 8. Near the root of f_{10}, the forward and backward candidates have markedly different local accuracy, so the fixed weight $w = 0.50$ produces a combination dominated by the less accurate candidate, inflating the effective error constant and slowing convergence. The adaptive weight detects this disparity through the residuals and shifts the concentration toward the more accurate candidate, recovering near-optimal second-order behaviour.

For functions where both methods converge quickly, such as f_1 and f_9, the constant-weight and adaptive methods require the same number of iterations. This behaviour is consistent with Theorem 6 (iii). When $f'(\alpha) \approx 0$, the error constant $C \approx c_2$ regardless of w, so no additional gain is available through weight adaptation in such cases.

Function $f_{12}(x) = x^3 - 1$ provides a useful stress test because $f'(\alpha) = 3$ at the root $\alpha = 1$, violating the condition $0 < f'(\alpha) < 1$ assumed in Corollary 8. In this case, the forward Steffensen method deteriorates severely, requiring 175 iterations, while the backward Steffensen method converges in only 6. The hybrid requires 16 iterations and the adaptive variant 15, reflecting the difficulty that a large value of $|f'(\alpha)|$ creates for both hybrid variants. Importantly, the adaptive method does not diverge in any of the twenty cases, suggesting reasonable robustness even outside the range covered by the corollary.

The entries for f_{15} and f_{16} show that the improvement in iteration count is not uniform across basins of attraction. On f_{15} the constant-weight hybrid leaves the basin of the tabulated root and settles at a different real root, and on f_{16} both hybrid variants do so. Iteration counts obtained at different limit points are not directly comparable, so these three entries are excluded from the comparison of convergence speed. The behaviour indicates that the residual-based weight controls the rate of convergence once a basin has been entered but does not by itself determine which root is reached, so the choice of initial guess remains consequential for both schemes.

Regarding the computational order of convergence, the COC is estimated from the last three iterates with nonzero error using (10). The forward, backward and constant-weight hybrid methods attain $COC = 2.00$ on every function for which they reach the tabulated root, in agreement with Theorem 6. For the adaptive variant the picture is less uniform. Because the method satisfies the stopping criterion in two to five steps on most problems, the estimate is formed in the pre-asymptotic regime, where the local rate of error decrease can be either higher or lower than the eventual asymptotic rate. Values above 2 arise where an intermediate step is unusually accurate, as for f_3 and f_6, and values below 2 arise where the final step dominates the estimate.

Repeating the computation with a much tighter stopping tolerance, so that the asymptotic regime is entered before termination, returns $COC = 2.00$ for the adaptive variant on every function, which confirms the second-order behaviour established in Theorem 6. Entries marked as a dash correspond to cases in which the method does not reach the root listed in Table 2, so that no meaningful estimate against that root is available.

Three limitations of the present scheme deserve mention. First, the residual-based weight (6) involves division by $|f(F_n)| + |f(B_n)|$, a quantity that approaches zero near the root. The regularisation parameter $\varepsilon = 10^{-15}$ prevents numerical overflow but introduces a small bias in the final iterations. Second, the adaptive scheme requires two additional residual evaluations per iteration relative to the constant-weight method, which lowers the per-iteration efficiency index from 1.260 to 1.149. The numerical results indicate that the reduction in iteration count outweighs the extra function evaluations across the entire test set, but this advantage may not extend uniformly to every problem class. Third, the convergence analysis assumes a simple root with $f'(\alpha) \neq 0$, and the behaviour of the method at multiple roots, where the Steffensen candidates lose second-order convergence, falls outside the present scope.

5. CONCLUSIONS

A modified hybrid Steffensen method has been proposed in which the fixed convex combination weight is replaced by a residual-based adaptive weight that dynamically favours the candidate with the smaller residual while preserving the convex structure at each iteration. The construction requires no user-specified parameter beyond the regularisation constant that guards against numerical instability close to the root.

A rigorous convergence analysis has established that the forward and backward Steffensen candidates carry different second-order error coefficients, so that the asymptotic error constant of the hybrid method depends on both the combination weight and the derivative of the function at the root. The equal-weight choice yields an error constant equal to that of the classical Newton-Raphson method, independent of the value of the derivative at the root. For the adaptive variant, the residual-based weight reduces the effective error constant below the equal-weight level whenever the derivative at the root lies strictly between zero and one, without raising the convergence order.

Numerical experiments on twenty test functions of varied analytical type have confirmed convergence to the correct root in all cases for the adaptive method, including one example in which the backward Steffensen variant failed to converge within the iteration budget. The adaptive variant required fewer or equal iterations relative to the equal-weight hybrid throughout the test set, with the largest reduction occurring on a cubic polynomial example where the equal-weight method required thirty-eight iterations against five for the adaptive variant. The per-iteration cost of the adaptive scheme is higher than that of the equal-weight hybrid because of the two additional residual evaluations needed to update the weight, yet the total number of function evaluations remains competitive because of the substantial reduction in the number of iterations.

Future work will extend the convergence analysis to multiple roots, investigate adaptive selection of the regularisation parameter to improve stability in later iterations, and generalise the method to systems of nonlinear equations. Comparative studies against higher-order Steffensen variants and methods with memory under common benchmarks would also be useful in clarifying the practical scope of the proposed scheme.

References

[1] I. Bate, K. Senapati, S. George, I. K. Argyros, M. I. Argyros, Convergence analysis of Jarratt-like methods for solving nonlinear equations for thrice-differentiable operators. AppliedMath 5(2) (2025) 38.

[2] H. M. S. Bawazir, Ninth and twelfth-order iterative methods for roots of nonlinear equations. Journal of Mathematical Analysis and Modeling 6(1) (2025) 35-45.

[3] G. Candelario, A. Cordero, J. R. Torregrosa, M. P. Vassileva, Generalized conformable fractional Newton-type method for solving nonlinear systems. Numerical Algorithms 93 (2023) 1171-1208.

[4] J. Dzunic, M. S. Petkovic, A cubically convergent Steffensen-like method for solving nonlinear equations. Applied Mathematics Letters 25(11) (2012) 1881-1886.

[5] H. Eskandari, Hybrid Steffensen's method for solving nonlinear equation. Applied Mathematics 13(11) (2022) 745-752.

[6] Y. H. Geum, Derivation of a numerical method with free second-order derivatives. International Journal of Pure Mathematics 9 (2022) 7-9.

[7] S. Karri, S. K. Vishwakarma, R. Sharma, G. Kaushik, A. Bhatt, R. Saxena, Technical indicator-based stock price forecasting using an extreme learning machine with deterministic weight adjustment. Proceedings of the 2025 IEEE 5th International Conference on ICT in Business Industry and Government (ICTBIG 2025).

[8] Z. Liu, Q. Zheng, A one-step Steffensen-type method with super-cubic convergence for solving nonlinear equations. Procedia Computer Science 29 (2014) 1870-1875.

[9] Z. Liu, Q. Zheng, P. Zhao, A variant of Steffensen's method of fourth-order convergence and its applications. Applied Mathematics and Computation 216(7) (2010) 1978-1983.

[10] M. Matinfar, M. Aminzadeh, Three-step iterative methods with eighth-order convergence for solving nonlinear equations. Journal of Interpolation and Approximation in Scientific Computing 2013 (2013) 1-11.

[11] P. K. Parida, D. K. Gupta, A cubic convergent iterative method for enclosing simple roots of nonlinear equations. Applied Mathematics and Computation 187(2) (2007) 1544-1551.

[12] M. Turkyilmazoglu, A simple algorithm for high order Newton iteration formulae and some new variants. Hacettepe Journal of Mathematics and Statistics 49(1) (2020) 425-438.

[13] M. Usman, S. Haq, A. Jan, A new iterative multi-step method for solving nonlinear equations. MethodsX 14 (2025) 103178.

[14] P. C. N. Verheijen, D. Goswami, M. Lazar, SuperADMM: solving quadratic programs faster with dynamic weighting ADMM. Proceedings of the 29th International Conference on System Theory, Control and Computing (ICSTCC 2025) 569-575.

[15] S. Weerakoon, T. G. I. Fernando, A variant of Newton's method with accelerated third-order convergence. Applied Mathematics Letters 13(8) (2000) 87-93.

[16] Q. Yang, J. Liu, Z. Wu, S. He, A fusion algorithm based on whale and grey wolf optimization algorithm for solving real-world optimization problems. Applied Soft Computing 146 (2023) 110701.

[17] I. F. Yumanova, One specification of Steffensen's method for solving nonlinear operator equations. Vestnik Udmurtskogo Universiteta. Matematika. Mekhanika. Komp'yuternye Nauki 26(4) (2016) 579-590.

[18] F. Zafar, A. Cordero, S. Mujtaba, J. R. Torregrosa, A hybrid Steffensen-genetic algorithm for finding multi-roots of nonlinear equations and applications to biomedical engineering. Algorithms 18(9) (2025) 582.

[19] Q. Zheng, J. Wang, P. Zhao, L. Zhang, A Steffensen-like method and its higher-order variants. Applied Mathematics and Computation 214(1) (2009) 10-16.

[20] Q. Zheng, P. Zhao, L. Zhang, W. Ma, Variants of Steffensen-secant method and applications. Applied Mathematics and Computation 216(12) (2010) 3486-3496.

[21] Q. Zheng, F. Huang, X. Guo, X. Feng, Doubly-accelerated Steffensen's methods with memory and their applications on solving nonlinear ODEs. Journal of Computational Analysis and Applications 15(5) (2013) 886-891.

[22] L. Zhu, T. Niu, M. Petrongolo, Iterative CT reconstruction via minimizing adaptively reweighted total variation. Journal of X-Ray Science and Technology 22(2) (2014) 227-240.

[23] Z. Zhu, A. Zhou, D. He, S. Wang, A novel hybrid method based on genetic algorithms and neural networks. Proceedings of the 2011 International Conference on Network Computing and Information Security (NCIS 2011), Vol. 2 (2011) 20-23.