mathematics

MDPI

Article

A Two-Dimensional Variant of Newton's Method and a Three-Point Hermite Interpolation: Fourth- and Eighth-Order Optimal Iterative Schemes

Chein-Shan Liu [1], Essam R. El-Zahar [2,3] and Chih-Wen Chang [4,*]

[1] Center of Excellence for Ocean Engineering, National Taiwan Ocean University, Keelung 202301, Taiwan; csliu@ntou.edu.tw
[2] Department of Mathematics, College of Sciences and Humanities in Al-Kharj, Prince Sattam bin Abdulaziz University, Alkharj 11942, Saudi Arabia; er.elzahar@psau.edu.sa
[3] Department of Basic Engineering Science, Faculty of Engineering, Menofia University, Shebin El-Kom 32511, Egypt
[4] Department of Mechanical Engineering, National United University, Miaoli 36063, Taiwan
* Correspondence: cwchang@nuu.edu.tw

Abstract: A nonlinear equation $f(x) = 0$ is mathematically transformed to a coupled system of quasi-linear equations in the two-dimensional space. Then, a linearized approximation renders a fractional iterative scheme $x_{n+1} = x_n - f(x_n)/[a + bf(x_n)]$, which requires one evaluation of the given function per iteration. A local convergence analysis is adopted to determine the optimal values of a and b. Moreover, upon combining the fractional iterative scheme to the generalized quadrature methods, the fourth-order optimal iterative schemes are derived. The finite differences based on three data are used to estimate the optimal values of a and b. We recast the Newton iterative method to two types of derivative-free iterative schemes by using the finite difference technique. A three-point generalized Hermite interpolation technique is developed, which includes the weight functions with certain constraints. Inserting the derived interpolation formulas into the triple Newton method, the eighth-order optimal iterative schemes are constructed, of which four evaluations of functions per iteration are required.

Keywords: nonlinear equation; two-dimensional approach; fractional iterative scheme; modified derivative-free Newton method; quadratures; fourth-order optimal iterative scheme; three-point generalized Hermite interpolation; eighth-order optimal iterative scheme

MSC: 41A25; 65D99; 65H05

check for updates

Citation: Liu, C.-S.; El-Zahar, E.R.; Chang, C.-W. A Two-Dimensional Variant of Newton's Method and a Three-Point Hermite Interpolation: Fourth- and Eighth-Order Optimal Iterative Schemes. *Mathematics* **2023**, *11*, 4529. https://doi.org/10.3390/math11214529

Academic Editor: Ioannis K. Argyros

Received: 10 October 2023
Revised: 28 October 2023
Accepted: 1 November 2023
Published: 3 November 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the paper, we develop several novel iterative schemes to solve the nonlinear equation:

$$f(x) = 0, \tag{1}$$

which is frequently occurred in engineering and scientific applications. For solving Equation (1),

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2}$$

is a well-known Newton iterative method. A lot of methods were modified from the Newton iterative method [1–8], and they are effective to solve nonlinear equations. We are going to replace $f'(x_n)$ in the denominator of Equation (2) by a linear function of $f(x_n)$, like as $a + bf(x_n)$ for some constants a and b. In doing so, a major drawback of the sensitivity to the initial guess of the Newton iterative method can be avoided, and at the same time, a major advantage is that $f'(x_n)$ is no longer needed.

The iterative schemes in [2,9–14] were based on the quadratures, which are of two-step-type iterative schemes with third-order convergence, needing the first Newton step to generate a trial solution, and then a correction at the second step by some quadrature rules. Our fractional iterative scheme is of the one-step type and also of the third-order convergence, which saves much computation of function per iteration.

Weerakoon and Fernando [2] resorted to a trapezoidal quadrature rule to derive an arithmetic mean Newton method with third-order convergence of the iterative scheme. After that, third-order iterative schemes based on different quadrature methods were developed in [9–12,15,16], of which the evaluations of $[f(x_n), f'(x_n), f'(\hat{x}_n)]$ with $\hat{x}_n = x_n - f(x_n)/f'(x_n)$ are required per iteration. They have the same order $p = 3$ and have the same efficiency index (E.I.)$= p^{1/m} = 1.44225$. However, the optimal order and efficiency index of the iterative scheme based on $[f(x_n), f'(x_n), f'(\hat{x}_n)]$ are $p = 2^{m-1} = 4$ and E.I.=1.5874, according to the conjecture of Kung and Traub [17].

For the fourth-order optimal iterative scheme (FOIS) with $[f(x_n), f'(x_n), f(\hat{x}_n)]$ performing at each iteration, there were many methods [1,17–29]. Recently, Liu and Liu [30] derived a double-weight function technique to derive the FOIS. The iterative schemes using difference modifications of Potra-Ptak's method with optimal fourth and eighth orders of convergence were performed by Cordero et al. [31]. However, the two-step FOIS based on $[f(x_n), f'(x_n), f'(\hat{x}_n)]$ is not yet developed. We are going to propose a simple method based on generalized quadratures to derive the FOIS, which is the optimal combination of two third-order iterative schemes to be developed in the paper.

Besides the quadratures and the finite difference approximation as used in the Ostrowski method and its modifications in [32–36], the function interpolation technique is often used to generate the high-order iterative scheme. Data interpolation is a mathematical process to construct the interpolant from the given data, of which the differential at each point is inserted to set up the Hermite interpolant, which was used in higher-order iterative schemes [30,37–43]. Also the three-point generalized Hermite interpolation techniques, and a new class of three-step eighth-order optimal iterative schemes (EOIS) is constructed in the paper, which involves four evaluations of functions that are optimal in the sense of Kung and Traub.

Previously, Zhanlav et al. [44] used the generating function method for constructing new EOIS iterations, by which our technique is quite different.

The scalar equation obtained in the engineering application is usually an implicit function of the unknown variable. For instance, the target equation used in the shooting method to solve the nonlinear boundary value problem is an implicit equation with $f(x)$ being an implicit function of x, and under this situation, it is hard to obtain the derivative term $f'(x)$. When the Newton iterative method cannot be applied to solve this sort problem, the proposed fractional iterative schemes have an advantage to solve this type of problem without using $f'(x)$.

The paper is organized as follows. A two-dimensional variant of Newton's iterative method is developed in Section 2. We derive a fractional iterative scheme, whose convergence criterion is proven. The convergence behavior analysis of the fractional iterative scheme is carried out in Section 3. We verify the performance of the proposed iterative schemes in Section 4 by computing several numerical examples. In Section 5, we combine the fractional iterative scheme to the quadrature methods to generate the FOIS. We reduce the fractional iterative schemes to some derivative-free iterative schemes in Section 6, and the convergence is identified. The Hermite interpolation is introduced in Section 7, and a three-point interpolation formula is derived. The results are used in Section 8 to derive the three-point generalized Hermite interpolations. In Section 9, we construct the EOIS by using the weight functions obtained from the three-point generalized Hermite interpolations, and examples are given. Finally, we draw the conclusions in Section 10. The abbreviations are listed in the Abbreviations.

2. Two-Dimensional Generalization of Newton's Method

To motivate the present study, we begin with

$$f'(x_n) = f'(r) + f''(r)(x_n - r) + \cdots, \tag{3}$$
$$f(x_n) = f'(r)(x_n - r) + \cdots, \tag{4}$$

where r is a simple solution of $f(x) = 0$ with $f(r) = 0$ and $f'(r) \neq 0$. Inserting Equation (3) for $f'(x_n)$ into Equation (2) and using $x_n - r = f(x_n)/f'(r)$ derived from Equation (4) with higher-order terms being neglected yields

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(r) + \frac{f''(r)}{f'(r)} f(x_n)}. \tag{5}$$

This iterative scheme is a variant near to the Newton iterative scheme (2). We will prove that the iterative scheme (5), like Equation (2), is quadratically convergent as to be stated by Theorem 3 in Section 3. Below we will derive an iterative scheme with a similar form to Equation (5), but it is cubically convergent, rather than the quadratic convergence of Equation (5).

When a new variable is defined by

$$y := f(x) = g(x)x, \tag{6}$$

where we suppose that $f(x)$ can be decomposed as $f(x) = g(x)x,\ x \neq 0$, from Equation (6) the following identity holds:

$$y - g(x)x = 0. \tag{7}$$

By Equations (1) and (7), we have

$$ax + y = ax, \tag{8}$$
$$[w(x) - 1]g(x)x + y = w(x)g(x)x, \tag{9}$$

where a is an accelerating parameter and $w(x)$ is a splitting function to be discussed below. While ax is added on both sides of the equation $y = 0$ in Equation (8), we add $w(x)g(x)x$ on both sides of Equation (7) to render Equation (9). Herein, the problem of finding the solution of Equation (1) is mathematically transformed to a coupled system of quasi-linear Equations (8) and (9) in the two-dimensional space (x, y).

The splitting technique of $-g(x)x = -(1 - w + w)g(x)x$ in Equation (9) is used to solve Equation (1) in [45]. Then, Liu et al. [46] proposed a derivative-free iterative scheme using $f(x) = g(x)x + Ax - B = 0$. We further carry out a theoretical analysis in the two-dimensional space directly for $f(x) = g(x)x = 0$.

When (x_n, y_n) is obtained at the nth step, the linearizations around x_n for Equations (8) and (9) are

$$ax + y = ax_n, \tag{10}$$
$$(w_n - 1)g_n x + y = w_n g_n x_n, \tag{11}$$

which are two-dimensional linear system for (x, y). We take $g_n = g(x_n)$ and $w_n = w(x_n)$ for shorting the notations.

From Equations (10) and (11), we can obtain (x_{n+1}, y_{n+1}) at the next step by

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & 1 \\ (w_n - 1)g_n & 1 \end{bmatrix}^{-1} \begin{bmatrix} ax_n \\ w_n g_n x_n \end{bmatrix},$$

which renders

$$x_{n+1} = \frac{ax_n - w_n g_n x_n}{a + (1 - w_n)g_n}, \tag{12}$$

$$y_{n+1} = \frac{ag_n x_n}{a + (1 - w_n)g_n} = \frac{ay_n}{a + (1 - w_n)g_n}. \tag{13}$$

Both the nominator and denominator on the right-hand side of Equation (12) are multiplied by x_n, and using Equation (6), it can be refined to

$$x_{n+1} = x_n - \frac{f(x_n)x_n}{ax_n + (1 - w_n)f(x_n)}. \tag{14}$$

Remark 1. *Mathematically, after adding $ax - w(x)f(x)$ on both sides, Equation (1) is equivalent to*

$$ax + f(x) - w(x)f(x) = ax - w(x)f(x),$$

which is then multiplied by x,

$$[ax + f(x) - w(x)f(x)]x = ax^2 - w(x)f(x)x.$$

If x_n is already known, we can seek the next x_{n+1} by

$$x_{n+1} = \frac{ax_n^2 - w(x_n)f(x_n)x_n}{ax_n + f(x_n) - w(x_n)f(x_n)} = x_n - \frac{f(x_n)x_n}{ax_n + f(x_n) - w(x_n)f(x_n)}. \tag{15}$$

Upon taking the same notation with $w_n = w(x_n)$, Equation (15) goes back to Equation (14). However, this one-dimensional approach cannot generate Equation (13); without setting the problem $f(x) = 0$ in the two-dimensional space (x, y) as carried out in Equations (8) and (9), it is hard to determine a and $w(x)$, and this proves Theorem 1 given below.

We are going to show that without resorting on the derivative term, the third-order convergence of the iterative scheme can also be realized. Our aim is reducing the number of function evaluations to one per iteration, without using the derivative term to maintain the order of convergence to be three.

Letting

$$1 - w_n = bx_n, \tag{16}$$

where b is a constant, we can cancel x_n in the fractional term of Equation (14), and with the aid of Equation (16), we can achieve

$$x_{n+1} = x_n - \frac{f(x_n)}{a + bf(x_n)}, \tag{17}$$

which, including two constant parameters a and b, is a novel iterative scheme to solve Equation (1). For use in the later, the iterative scheme (17) is called a fractional iterative scheme.

Theorem 1. *For solving $f(x) = 0$, the iterative scheme (17) is convergent, if*

$$f^2(x_n) + 2abf(x_n) > 0. \tag{18}$$

A finer criterion is

$$\frac{bf(x_n)}{a} > 0, \quad or \quad \frac{bf(x_n)}{a} < -2. \tag{19}$$

Proof. In Equation (13), y_{n+1} is replaced by $f(x_{n+1})$ and y_n is replaced by $f(x_n)$. In view of Equations (6) and (16), we have

$$f(x_{n+1}) = \frac{af(x_n)}{a + bf(x_n)}. \tag{20}$$

Let

$$\xi_n := \frac{a}{a + bf(x_n)}$$

be a contraction factor. By Equation (18), we have

$$f^2(x_n) + 2abf(x_n) > 0 \Rightarrow |\xi_n| < 1, \tag{21}$$

which implies a strictly monotonically decreasing sequence of

$$|f(x_{n+1})| < |f(x_n)|. \tag{22}$$

Hence, the absolute convergence of the iterative scheme (17) was proved. Equation (20) can be written as

$$f(x_{n+1}) = \frac{f(x_n)}{1 + \frac{bf(x_n)}{a}}. \tag{23}$$

If the criterion in Equation (19) is satisfied, from Equation (23), we can derive the inequality in Equation (22). The proof is completed. □

Remark 2. *Although we have made a decomposition of $f(x) = xg(x)$ in Equation (6), the final results in Equations (17) and (20) are independent to $g(x)$. However, the decomposition technique in Equation (6) can help us to derive the two-dimensional approach to the Newton method and the fractional iterative scheme.*

3. Convergence Analysis of Fractional Iterative Scheme

The convergence analysis of Equation (17) is given below.

Theorem 2. *The iterative scheme (17) for solving $f(x) = 0$ has third-order convergence, with the parameters given by*

$$a = f'(r),\ b = \frac{f''(r)}{2f'(r)}. \tag{24}$$

Proof. For the proof of convergence, let r be a simple solution of $f(x) = 0$, i.e., $f(r) = 0$ and $f'(r) \neq 0$. Thus, by giving

$$e_n = x_n - r,$$

it follows that

$$e_{n+1} = e_n + x_{n+1} - x_n, \tag{25}$$
$$f(x_n) = f'(r)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + \cdots], \tag{26}$$

where

$$c_k := \frac{f^{(k)}(r)}{k!f'(r)},\ k = 2, 3, \ldots.$$

Inserting Equation (26) into Equation (17) yields

$$\frac{f(x_n)}{a + bf(x_n)} = \frac{e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + \cdots}{1 + be_n + bc_2e_n^2 + bc_3e_n^3 + \cdots} = e_n + D_2e_n^2 + D_3e_n^3 + D_4e_n^4 + \cdots, \tag{27}$$

where we have used the first one in Equation (24), and D_2, D_3 and D_4 are given by

$$\begin{aligned} D_2 &= c_2 - b, \\ D_3 &= c_3 - 2c_2 b + b^2, \\ D_4 &= 2b^2 c_2 - bc_3 + c_2(b^2 - bc_2) - c_3 b + c_4 - b^3. \end{aligned} \tag{28}$$

Inserting Equation (27) into Equation (17) and using Equation (25) yields

$$e_{n+1} = e_n - e_n - D_2 e_n^2 - D_3 e_n^3 - D_4 e_n^4 - \cdots = -D_2 e_n^2 - D_3 e_n^3 - D_4 e_n^4 - \cdots . \tag{29}$$

If $b = c_2$ holds as that given in second one in Equation (24), we have

$$D_2 = 0, \;\; D_3 = c_3 - c_2^2, \;\; D_4 = c_2^3 - 2c_2 c_3 + c_4,$$

and at the same time, Equation (29) reduces to

$$e_{n+1} = (c_2^2 - c_3) e_n^3 + \mathcal{O}(e_n^4). \tag{30}$$

Equation (30) indicates the third-order convergence. □

Theorem 3. *The iterative scheme (5) for solving $f(x) = 0$ has second-order convergence.*

Proof. Upon comparing to Equation (17), the parameters in Equation (5) are given by

$$a = f'(r), \; b = \frac{f''(r)}{f'(r)}.$$

Inserting b into D_2 in Equation (28) yields

$$D_2 = c_2 - \frac{f''(r)}{f'(r)} = \frac{f''(r)}{2f'(r)} - \frac{f''(r)}{f'(r)} = -\frac{f''(r)}{2f'(r)} \neq 0.$$

By Equation (29),

$$e_{n+1} = e_n - e_n - D_2 e_n^2 - D_3 e_n^3 - D_4 e_n^4 - \cdots = \frac{f''(r)}{2f'(r)} e_n^2 + \mathcal{O}(e_n^3)$$

proves this theorem. □

In practice, the iterative scheme (5) is a variant of the Newton iterative scheme (2). Both orders of convergence are two.

Notice that the Newton method (2) is a single-point second-order optimal iterative scheme, with two function operations of $f(x_n)$ and $f'(x_n)$. Halley [47] derived the following extension of the Newton method to a third-order iterative scheme:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - f(x_n)\frac{f''(x_n)}{2f'(x_n)}}. \tag{31}$$

However, because it needs three function operations on $f(x_n)$, $f'(x_n)$ and $f''(x_n)$, it is not an optimal iterative scheme. Besides the Halley method, there are many two-point iterative schemes which are of third-order convergence. Liu and Lee [48] generalized many quadrature-type third-order iterative schemes to

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \;\; \eta_n = \frac{f'(x_n)}{f'(y_n)} - 1, \\ x_{n+1} &= x_n - W(\eta_n)\frac{f(x_n)}{f'(x_n)}, \;\; W(0) = 1, \; W'(0) = -\frac{1}{2}, \; |W''(0)| < \infty. \end{aligned} \tag{32}$$

Based on three function operations of $f(x_n)$, $f'(x_n)$ and $f'(y_n)$, Equation (32) is not an optimal iterative scheme. It is interesting that upon comparing Equation (31) to Equation (17), these two iterative schemes are the same if we take $a = f'(x_n)$ and $b = -f''(x_n)/(2f'(x_n))$. But merely with a and b given by Equation (24), the iterative scheme (17) is of third-order convergence. It must be emphasized that we do not need $f'(x_n)$, $f''(x_n)$ and two-point operation to achieve the third-order convergence. Therefore, the key issue of Equations (17) and (24) is that we need to give a precise estimation of a and b, without using the information from $f'(r)$ and $f''(r)$.

4. Numerical Verifications

Some examples are used to evaluate the iterative scheme (17), which is subjected to the convergence criteria:

$$|x_{n+1} - x_n| < \varepsilon \text{ and } |f(x_n)| < \varepsilon.$$

We fix $\varepsilon = 10^{-15}$ for all numerical tests. The numerically computed order of convergence (COC) is defined in [2]:

$$\text{COC} := \frac{\ln|(x_{n+1} - r)/(x_n - r)|}{\ln|(x_n - r)/(x_{n-1} - r)|}, \tag{33}$$

where $f(r) = 0$.

The presently computed results are compared to those obtained by the Newton method (NM), the Halley method (HM) [47] in Equation (31) and the method of Li (LM) [18]:

$$\hat{x}_n = x_n - \frac{f(x_n)}{f'(x_n)}, \quad x_{n+1} = x_n - \frac{f(x_n)[f(x_n) - f(\hat{x}_n)]}{f'(x_n)[f(x_n) - 2f(\hat{x}_n)]}. \tag{34}$$

The orders of convergence for the NM, HM and LM are two, three and four, respectively.

We first use the following example:

$$f(x) = x^3 - \frac{1}{2}x^2 - \frac{7}{2}x + 3 = 0, \tag{35}$$

to present the monotonically decreasing sequence of $f(x_n)$, which is generated by the method in Equation (17). The parameters a and b are specified, not that given by Equation (24).

There are three solutions $r = -2$, $r = 1$ and $r = 1.5$ of Equation (35). We consider four cases:

(a) $x_0 < 0$, $f(x_0) < 0$, $a > 0$, $b < 0$,
(b) $x_0 < 0$, $f(x_0) > 0$, $a > 0$, $b < 0$,
(c) $x_0 > 0$, $f(x_0) < 0$, $a < 0$, $b < 0$,
(d) $x_0 > 0$, $f(x_0) > 0$, $a > 0$, $b > 0$.

We take $x_0 = -3$, $a = 10.5$ and $b = -0.619$ for (a); $x_0 = -1$, $a = 10.5$ and $b = -0.619$ for (b); $x_0 = 1.1$, $a = -1.5$ and $b = -1.66$ for (c); $x_0 = 2.5$, $a = 1.75$ and $b = 1.8$ for (d). Cases (a) and (b) tend to the solution $r = -2$; case (c) tends to $r = 1$; and case (d) tends to $r = 1.5$.

Due to the monotonically decreasing sequence of $f(x_n)$, all the COCs are near to the third-order and they converge very fast. In the last column of Table 1, we list the NIs by using the LM of [18]. Starting from the same initial values, for the first two cases, the scheme (17) is convergent slightly faster than the LM. As mentioned by Li [18], the iterative scheme (34) requires two evaluations of the function and one first-order derivative per iteration. Therefore, the scheme (17) with one evaluation of the function saves much of the computational cost.

Table 1. For Equation (35), showing the monotonically decreasing sequence of $f(x_n)$ with respect to the number n of steps and the COC for different cases and listing the number of iterations (NI) of present method. The last column is the NI of LM.

n	0	1	2	3	4	5	COC	NI	LM
(a)	-18	-1.9558	-1.3362×10^{-2}	-5.4058×10^{-9}	0	$\times$	3.01	4	6
(b)	5	2.7588	0.1148	4.1074×10^{-6}	0	$\times$	3.03	4	6
(c)	-0.1240	-6.2336×10^{-3}	-5.5696×10^{-7}	-1.7764×10^{-15}	0	$\times$	2.94	4	4
(d)	6.75	2.0947	0.3115	2.6840×10^{-4}	-1.9952×10^{-8}	0	2.98	5	5

Other test examples are given by

$$f_1(x) = \exp(x^2 + 7x - 30) - 1, \tag{36}$$
$$f_2(x) = x^3 + 4x^2 - 10, \tag{37}$$
$$f_3(x) = xe^{x^2} - \sin^2 x + 3\cos x + 5, \tag{38}$$
$$f_4(x) = |x^2 - 9|. \tag{39}$$

Table 2 lists x_0, a and b and the NIs for different methods. We can observe that the present iterative scheme converges faster than the NM and HM. NM and HM are not good for the solution of $f_2(x) = 0$ with a worse initial guess $x_0 = -0.3$, and cannot be applied to solve $f_4(x) = 0$ in Equation (39).

Table 2. For $f_i(x) = 0,\ i = 1, \ldots, 4$, listing x_0, a and b and the NIs of present method. The last two columns are the NIs of NM and HM.

$f = 0$	x_0	a	b	Present NI	NI (NM)	NI (HM)
$f_1 = 0$	4	13	2	7	20	11
$f_2 = 0$	-0.3	16.5134	0.49	6	55	142
$f_3 = 0$	-2	20.307	-1	6	9	6
$f_4 = 0$	3.2	6	0.2	4	$\times$	$\times$

5. Fourth-Order Optimal Iterative Schemes

Now, we propose some new FOIS by a constantly weighting combination of the third-order iterative schemes from Equations (17) and (24), as well as the following one. Before that, we cite the following result [48].

Lemma 1. *The following two-step iterative scheme has third-order convergence:*

$$x_{n+1} = x_n - W(\eta_n)\frac{f(x_n)}{f'(x_n)}, \tag{40}$$

where W satisfies

$$W(0) = 1,\ W'(0) = -\frac{1}{2},\ |W''(0)| < \infty.$$

Here,

$$\hat{x}_n = x_n - \frac{f(x_n)}{f'(x_n)},\ \ \eta_n := \frac{f'(\hat{x}_n)}{f'(x_n)} - 1.$$

The error equation reads as

$$e_{n+1} = \left(2[1 - W''(0)]c_2^2 + \frac{c_3}{2}\right)e_n^3 + \mathcal{O}(e_n^4). \tag{41}$$

Proof. The proof can refer to [48]. □

Theorem 4. *The following iterative scheme as an optimal combination of Equations (17) and (40):*

$$x_{n+1} = x_n - \frac{w_1 f(x_n)}{a + bf(x_n)} - w_2 W(\eta_n)\frac{f(x_n)}{f'(x_n)} \tag{42}$$

is of fourth-order convergence, where $W(0) = 1$, $W'(0) = -1/2$, $W''(0) = \alpha$, *and*

$$w_1 = \frac{1}{3},\ w_2 = \frac{2}{3},\ \alpha = \frac{5}{4}. \tag{43}$$

Proof. The combination of Equations (17) and (40) is given by Equation (42), whose weighting factors w_1 and w_2 are subjected to

$$w_1 + w_2 = 1. \tag{44}$$

Then, we consider the weighting combination of the error equations in Equations (30) and (41), such that the combined coefficient preceding e_n^3 is zero:

$$w_1(c_2^2 - c_3) + w_2\left(2[1-\alpha]c_2^2 + \frac{c_3}{2}\right) = 0 \Rightarrow w_1(c_2^2 - c_3) - \frac{w_2}{2}[(4\alpha - 4)c_2^2 - c_3] = 0. \tag{45}$$

Equation (45) leads to

$$4\alpha - 4 = 1,\ \ w_1 - \frac{w_2}{2} = 0. \tag{46}$$

Solving Equations (44) and (46), we can derive Equation (43). Thus, the error equation of the optimally combined iterative scheme in Equation (42) is

$$e_{n+1} = \mathcal{O}(e_n^4).$$

This completes the proof of Theorem 4. □

As an application of Theorem 4, we consider [49]:

$$\begin{cases} x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)^{\gamma-1} + f'(\hat{x}_n)^{\gamma-1}]}{f'(x_n)^{\gamma} + f'(\hat{x}_n)^{\gamma}}, \\ e_{n+1} = \left[\gamma c_2^2 + \frac{c_3}{2}\right]e_n^3 + \mathcal{O}(e_n^4), \end{cases}$$

which in the form of Equation (40) leads to

$$W(\eta_n) = \frac{1 + (\eta_n + 1)^{\gamma-1}}{1 + (\eta_n + 1)^{\gamma}}.$$

When we take $\gamma = -1/2$, the conditions $W(0) = 1$, $W'(0) = -1/2$, $W''(0) = 5/4$ in Theorem 4 are satisfied. In Table 3, we solve Equation (35) by using the FOIS (42), and list the results for three different solutions $r = -2, 1, 1.5$, which show large values of the COC.

Table 3. For Equation (35), listing x_0, NIs and COCs of the FOIS (42) for three solutions.

r	x_0	NI	COC
-2	-2.5	4	3.828
1	0.8	4	3.694
1.5	1.4	4	4.278

Let $\alpha = W''(0)$ be a parameter. With $W(\eta) = 1 - \eta/2 + \alpha\eta^2/2$ in Equation (40), we have

$$\begin{cases} x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\left(\frac{3}{2} - \frac{f'(\hat{x}_n)}{2f'(x_n)} + \frac{\alpha[f'(\hat{x}_n)-f'(x_n)]^2}{2f'(x_n)^2}\right), \\ e_{n+1} = \left(2[1-\alpha]c_2^2 + \frac{c_3}{2}\right)e_n^3 + \mathcal{O}(e_n^4). \end{cases}$$

The best value of $\alpha = 5/4$ is chosen such that

$$x_{n+1} = x_n - \frac{f(x_n)}{3[a+bf(x_n)]} - \frac{2f(x_n)}{3f'(x_n)}\left(\frac{3}{2} - \frac{f'(\hat{x}_n)}{2f'(x_n)} + \frac{\alpha[f'(\hat{x}_n) - f'(x_n)]^2}{2f'(x_n)^2}\right) \tag{47}$$

is an FOIS.

In Table 4, we solve Equation (36) and list the results for the solution $r = 3$ with four different initial guesses: $x_0 = 3.1, 3.5, 4, 5$, which show large values of the COC.

Table 4. For Equation (36), listing x_0, NIs and COCs of the FOIS (47) for the solution $r = 3$.

x_0	3.1	3.5	4	5
NI	5	8	12	21
COC	3.325	3.811	3.741	3.339

6. Derivative-Free Iterative Schemes

In this section, we approximate a and b in Equation (24). With two initial guesses x_0 and $\tilde{x}_2$ satisfying $f(x_0)f(\tilde{x}_2) < 0$ to render $r \in (x_0, \tilde{x}_2)$, $\tilde{x}_1 = (x_0 + \tilde{x}_2)/2$ is taken. By a finite difference approximation of a and b, we take

$$a = \frac{f(\tilde{x}_2) - f(x_0)}{\tilde{x}_2 - x_0}, \quad b = \frac{1}{2a}\frac{f(\tilde{x}_2) - 2f(\tilde{x}_1) + f(x_0)}{(\tilde{x}_1 - x_0)^2} = \frac{2f(\tilde{x}_2) - 4f(\tilde{x}_1) + 2f(x_0)}{(\tilde{x}_2 - x_0)[f(\tilde{x}_2) - f(x_0)]}. \tag{48}$$

Inserting a and b in Equation (48) into Equation (17), we solve Equation (35) and the related data are tabulated in Table 5.

Table 5. For Equation (35), listing x_0, $\tilde{x}_2$, a and b, NIs and COCs of present method for three solutions: $r = -2, 1, 1.5$.

r	x_0	$\tilde{x}_2$	a	b	NI	COC
-2	-2.5	-1.5	10.75	-0.60645	11	1.0059
1	0.8	1.2	-1.46	-1.71233	11	1.01383
1.5	1.4	1.6	1.76	2.27273	8	1.00138

We solve Equations (36)–(39), and the related data are tabulated in Table 6.

Table 6. For $f_i(x) = 0$, $i = 1, \ldots, 4$, listing x_0, $\tilde{x}_2$, a and b, NIs and COCs of present method.

i	x_0	$\tilde{x}_2$	a	b	NI	COC
1	2.9	3.07	13.0621	5.96248	9	0.978064
2	0.9	1.3	11.67	0.282776	9	0.9819139
3	-1.25	-1.2	21.43125	-1.5275414	11	1.009
4	3.2	3.3	6.5	0.1538462	14	1.11766

Remark 3. *For the solution of $f_4(x) = 0$, it does not exist x_0 and $\tilde{x}_2$ such that $f_4(x_0)f_4(\tilde{x}_2) < 0$, due to $f_4(x) \geq 0$. However, we place x_0 and $\tilde{x}_2$ on the right-side of the solution and the present iterative scheme is applicable to find the solution with 14 iterations, as shown in Table 6.*

If the curve of $f(x)$ vs. x is available, we can observe a rough position of the solution r, and then the slope $f'(r)$ and the curvature $f''(r)$ can be estimated roughly. Intuitively, we can estimate a and b by the slope and curvature. In order to maintain the fast convergence, we must choose $\tilde{x}_0$ and $\tilde{x}_2$ quite close to the solution r, such that

$$\tilde{a} = \frac{f(\tilde{x}_2) - f(\tilde{x}_0)}{\tilde{x}_2 - \tilde{x}_0}, \quad \tilde{b} = \frac{1}{2\tilde{a}} \frac{f(\tilde{x}_2) - 2f(\tilde{x}_1) + f(\tilde{x}_0)}{(\tilde{x}_1 - \tilde{x}_0)^2} \tag{49}$$

are very close to a and b in Equation (24), where $\tilde{x}_1 = (\tilde{x}_0 + \tilde{x}_2)/2$. For Equation (35) with the solution $r = -2$, if we take $\tilde{x}_0 = r - 0.01$ and $\tilde{x}_2 = r + 0.01$, NI is greatly reduced from 11 to 5 and COC=2.904; for $r = 1$, NI reduces to 6 and COC = 2.862; and for $r = 1.5$, NI reduces to 5 and COC = 2.936. Here, NI and COC are improved by comparing to that in Table 5.

A greed search such as the 2D golden section search algorithm in the given range can help us to obtain the optimal values of a and b for fast convergence. However, it would spend much more computations. Instead of a greed search in the plane (a, b), we discuss the influence of $\tilde{a}$ and $\tilde{b}$ in Equation (49) by giving $\tilde{x}_0 = r - c$ and $\tilde{x}_2 = r + c$. For different c, $\tilde{a}$ and $\tilde{b}$ are different. In Table 7, we list the results. Obviously, COC defined in Equation (33) is sensitive to the values of $\tilde{a}$ and $\tilde{b}$, when they approach to the optimal values, but NI does not have a large variation. When $\tilde{a}$ and $\tilde{b}$ tend to optimal values $a = 10.5$ and $b = -0.619047619$, COC = 2.9041 tends to the theoretical one with COC = 3, as listed in Table 1.

Table 7. For Equation (35) with the solution $r = -2$ and with the initial guess $x_0 = -2.5$, listing $\tilde{a}$, $\tilde{b}$, NIs and COCs of present method for different values of c.

c	$\tilde{a}$	$\tilde{b}$	NI	COC
0.1	10.51	−0.61845861	7	0.999994
0.05	10.5025	−0.61890026	6	1.084194
0.02	10.50004	−0.61902404	6	1.233496
0.011	10.500012	−0.61904049	5	2.90119
0.01	10.50001	−0.61904172	5	2.90404

Like that performed in Equations (48) and (17), we introduce a derivative-free modification of the Newton variant in Equation (5) to

$$x_{n+1} = x_n - \frac{f(x_n)}{A + Bf(x_n)}, \tag{50}$$

where

$$A = \frac{f(\tilde{x}_2) - f(x_0)}{\tilde{x}_2 - x_0}, \quad B = \frac{1}{A} \frac{f(\tilde{x}_2) - 2f(\tilde{x}_1) + f(x_0)}{(\tilde{x}_1 - x_0)^2} = \frac{4f(\tilde{x}_2) - 8f(\tilde{x}_1) + 4f(x_0)}{(\tilde{x}_2 - x_0)[f(\tilde{x}_2) - f(x_0)]}. \tag{51}$$

For Equations (36)–(39) solved by the first derivative-free Newton method (FDFNM), the related data are tabulated in Table 8.

Table 8. For $f_i(x) = 0$, $i = 1, \ldots, 4$, listing x_0, $\tilde{x}_2$, A and B, NIs and COCs of the FDFNM.

i	x_0	$\tilde{x}_2$	A	B	NI	COC
1	2.9	3.07	13.0621	11.92497	10	1.00286
2	0.9	1.3	11.67	0.565553	9	1.000512
3	−1.25	−1.2	21.43125	−3.055083	12	1.05017
4	3.2	3.3	6.5	0.30729631	14	1.00584

By neglecting the higher-order terms in Equations (2) and (3), we have

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(r) + f''(r)(x_n - r)}. \tag{52}$$

We have

$$f(x_n) = f(r) + f'(r)(x_n - r) + \frac{1}{2}f''(r)(x_n - r)^2 + \cdots. \tag{53}$$

By using $f(r) = 0$ and neglecting the higher-order terms, it follows from Equation (53) a quadratic equation for $x_n - r$:

$$f''(r)(x_n - r)^2 + 2f'(r)(x_n - r) - 2f(x_n) = 0.$$

Thus, we can derive

$$x_n - r = \frac{\sqrt{f'(r)^2 + 2f(x_n)f''(r)} - f'(r)}{f''(r)}. \tag{54}$$

Inserting Equation (54) into Equation (52), we can derive the second modified Newton method:

$$x_{n+1} = x_n - \frac{f(x_n)}{\sqrt{f'(r)^2 + 2f(x_n)f''(r)}}, \tag{55}$$

which is different from the first modified Newton method (5). Let

$$C = \frac{f(\tilde{x}_2) - f(x_0)}{\tilde{x}_2 - x_0}, \quad D = \frac{f(\tilde{x}_2) - 2f(\tilde{x}_1) + f(x_0)}{(\tilde{x}_1 - x_0)^2}, \tag{56}$$

and we can obtain the second derivative-free Newton method (SDFNM):

$$x_{n+1} = x_n - \frac{f(x_n)}{\sqrt{C^2 + 2Df(x_n)}}. \tag{57}$$

We employ the SDFNM to solve Equations (36)–(39), and the related data are tabulated in Table 9.

Table 9. For $f_i(x) = 0$, $i = 1, \ldots, 4$, listing x_0, $\tilde{x}_2$, A and B, NIs and COCs of the SDFNM.

i	x_0	$\tilde{x}_2$	C	D	NI	COC
1	2.9	3.07	13.0621	155.8141	8	0.9984
2	0.9	1.3	11.67	6.6	9	1.0022
3	−1.25	−1.2	21.43125	−65.4743	12	1.0311
4	3.2	3.3	6.5	2	14	0.98004

Upon comparing Tables 6, 8 and 9, the performance of the presented method in Equations (48) and (17), as well as the FDFNM in Equations (50) and (51) and the SDFNM in Equations (56) and (57), are almost the same.

As a practical application of the proposed iterative schemes, let us consider a nonlinear boundary value problem:

$$u''(y) = \frac{3}{2}u^2(y), \; y \in (0, 1), \tag{58}$$

$$u(0) = 4, \quad u(1) = 1. \tag{59}$$

An exact solution is

$$u(y) = \frac{4}{(y+1)^2}.$$

The conventional shooting method is assumed to be an unknown initial slope $u'(0) = x$, and we integrate Equation (58) with the initial conditions $u(0) = 4$ and $u'(0) = x$, which results an implicit equation $f(x) = u(1, x) - 1 = 0$ to be solved. The exact solution is $r = -8$.

We apply the fourth-order Runge–Kutta method to integrate Equation (58) with $N = 7500$ steps, and fix $x_0 = -9.5$ and $\tilde{x}_2 = -6$. By using Equations (48) and (17), NI is 16, the error of x is 2.842×10^{-14} and the maximum error of u is 5.33×10^{-15}. While we use Equations (50) and (51), NI is increased to 18, and the errors are the same.

7. Hermite Interpolation

To be the extensions of the one-step Newton method (2), there are, respectively, two-step and three-step methods of double Newton and triple Newton:

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)}, \end{cases} \quad \begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \frac{f(y_n)}{f'(y_n)}, \\ x_{n+1} = z_n - \frac{f(z_n)}{f'(z_n)}. \end{cases} \tag{60}$$

However, due to the low efficiency index (E.I.) = 1.414 of these iterative schemes, they are rarely used in the solution of nonlinear equations. Below, we will employ the generalized Hermite interpolation techniques to raise the values of E.I.

We fix

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{61}$$

$$\xi_n = \frac{f(y_n)}{f(x_n)}, \ \eta_n = \frac{f(z_n)}{f(y_n)}, \ \gamma_n = \frac{f(z_n)}{f(x_n)} = \xi_n \eta_n. \tag{62}$$

The Hermite function $H(x)$ for the interpolation of the data of a function $f(x)$ at two points x_n and y_n is such that

$$H(x_n) = f(x_n), \ H(y_n) = f(y_n), \ H'(x_n) = f'(x_n), \ H'(y_n) = f'(y_n). \tag{63}$$

If $H(x)$ is a polynomial to match these four conditions in Equation (63), it is at least a second-order function of x, denoted as $H_2(x)$. When y_n is computed from Equation (61), it is not independent to x_n; hence, there exists an Hermite interpolation formula to predict $f'(y_n)$ from the data $(f(x_n), f(y_n), f'(x_n))$. The two-point Hermite interpolation formula was generalized in [30], involving a weight function.

The second-order Hermite polynomial is constructed according to the Hermite interpolation conditions [30]:

$$H_2(x_n) = f(x_n), \ H_2(y_n) = f(y_n), \ H_2'(x_n) = f'(x_n).$$

Wang and Liu [36] derived

$$\begin{aligned} H_2(x) &= \frac{(x - y_n) f(x_n)}{x_n - y_n} - \frac{x - x_n}{(x_n - y_n)^2} [(x - y_n) f(x_n) - (x - x_n) f(y_n)] \\ &+ \frac{(x - x_n)(x - y_n)}{x_n - y_n} f'(x_n), \\ H_2'(x) &= \frac{2(x - x_n)}{(y_n - x_n)^2} [f(y_n) - f(x_n)] - \frac{2x - x_n - y_n}{y_n - x_n} f'(x_n). \end{aligned}$$

Then, it follows from $f'(y_n) = H_2'(y_n)$ and Equations (61) and (62) that

$$f'(y_n) = (1 - 2\xi_n)f'(x_n), \tag{64}$$

which expressed a certain two-point generalized Hermite interpolation.

Definition 1. *A two-point generalized Hermite interpolation of $f'(y_n)$ in terms of $f'(x_n)$ and ξ_n is depicted by*

$$f'(y_n) = h(\xi_n)f'(x_n), \tag{65}$$

where the weight function $h(\xi_n)$ satisfies

$$h(0) = 1,\ h'(0) = -2. \tag{66}$$

Equations (65) and (66) include Equation (64) as a special case.

If one replaces $f'(y_n)$ in the first equation in Equation (60) by that in Equation (65), it generates an FOIS [17,37,38]:

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = y_n - \frac{1}{h(\xi_n)}\frac{f(y_n)}{f'(x_n)}. \end{cases} \tag{67}$$

The E.I. of Equation (67) is now raised to E.I. = 1.587, which is better than E.I. = 1.414 of the double Newton method.

This fact encourages us also to replace $f'(z_n)$ in Equation (60) by the following three-point interpolation formula:

$$f'(z_n) = g(\xi_n, \eta_n, \gamma_n)f'(x_n), \tag{68}$$

such that the combination of Equations (60), (65) and (68) leads to

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \frac{1}{h(\xi_n)}\frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} = z_n - \frac{1}{g(\xi_n,\eta_n,\gamma_n)}\frac{f(z_n)}{f'(x_n)}. \end{cases} \tag{69}$$

With certain conditions on $h(\xi_n)$ and $g(\xi_n, \eta_n, \gamma_n)$, the E.I. of the iterative scheme (69) can be further raised to E.I. = 1.682. They are three-point EOISs.

8. Three-Point Generalized Hermite Interpolations

Using the third-order Hermite polynomial for the three-point (x_n, y_n, z_n) interpolation, Wang and Liu [36] and Petković [37] derived

$$f'(z_n) = -2f[x_n, y_n] + f[y_n, z_n] + 2f[x_n, z_n] + (y_n - z_n)f[y_n, x_n, x_n], \tag{70}$$

where

$$f[x_n, y_n] = \frac{f(x_n) - f(y_n)}{x_n - y_n},\ f[y_n, x_n, x_n] = \frac{f[y_n, x_n] - f'(x_n)}{y_n - x_n},$$

and $f[x_n, z_n]$ and $f[y_n, z_n]$ are defined by the same fashion.

From the first two equations in Equation (69) and Equation (62), we can derive the following divided differences:

$$\begin{aligned} &f[x_n, y_n] = (1-\xi_n)f'(x_n), \\ &f[y_n, z_n] = h(\xi_n)(1-\eta_n)f'(x_n),\ f[x_n, z_n] = \frac{h(\xi_n)(1-\gamma_n)}{h(\xi_n)+\xi_n}f'(x_n), \end{aligned} \tag{71}$$

$$f[y_n, x_n, x_n] = \frac{\xi_n}{f(x_n)}f'(x_n)^2,$$

$$f[z_n, x_n, x_n] = \frac{f[z_n, x_n] - f'(x_n)}{z_n - x_n} = \frac{[h(\xi_n)\gamma_n + \xi_n]h(\xi_n)}{[h(\xi_n)+\xi_n]^2 f(x_n)}f'(x_n)^2. \tag{72}$$

Using Equations (70), (71) and (72) and through some manipulations, we can derive

$$f'(z_n) = [A(\xi_n) - B(\xi_n)\eta_n - C(\xi_n)\gamma_n]f'(x_n), \tag{73}$$

where

$$A(\xi_n) := 2\xi_n + h(\xi_n) - \frac{2\xi_n}{h(\xi_n)+\xi_n} + \frac{\xi_n^2}{h(\xi_n)},\ B(\xi_n) = h(\xi_n),\ C(\xi_n) := \frac{2h(\xi_n)}{h(\xi_n)+\xi_n}. \tag{74}$$

Definition 2. *A three-point generalized Hermite interpolation of $f'(z_n)$ in terms of $f'(x_n)$ and $g(\xi_n, \gamma_n, \eta_n)$ is defined by Equations (73) and (74) as*

$$f'(z_n) = g(\xi_n, \eta_n, \gamma_n)f'(x_n) := [A(\xi_n) - B(\xi_n)\eta_n - C(\xi_n)\gamma_n]f'(x_n), \tag{75}$$

where the weight functions $A(\xi_n)$, $B(\xi_n)$ and $C(\xi_n)$ satisfy

$$A(0) = 1,\ A'(0) = -2,\ A''(0) = -2,\ A'''(0) = 0,\ \ B(0) = 1,\ B'(0) = -2,\ \ C(0) = 2. \tag{76}$$

For the two-point Hermite interpolation function $h(\xi_n) = 1 - 2\xi_n$ in Equation (64), by Equation (74), we can derive

$$A(\xi_n) = \frac{2-4\xi_n}{1-\xi_n} + \frac{\xi_n^2}{1-2\xi_n} - 1,\ B(\xi_n) = 1-2\xi_n,\ C(\xi_n) = \frac{2-4\xi_n}{1-\xi_n}. \tag{77}$$

Equation (77) is a special case of Equation (76). Unlike that in Equation (74), for the generalized interpolation in Equation (75), $g(\xi_n, \gamma_n, \eta_n) = A(\xi_n) - B(\xi_n)\eta_n - C(\xi_n)\gamma_n$ can be independent to $h(\xi_n)$.

The function $A(\xi)$ is subjected to four conditions, in which it is somewhat not easy to obtain $A(\xi)$. We attempt to construct it by a function with merely two conditions.

Lemma 2. *A function $A(\xi)$, with*

$$A(0) = a_0,\ A'(0) = a_1,\ A''(0) = a_2,\ A'''(0) = a_3, \tag{78}$$

can be obtained from

$$A(\xi) = a_0 + a_1\xi + \xi^2 F(\xi), \tag{79}$$

where

$$F(0) = \frac{a_2}{2},\ F'(0) = \frac{a_3}{6}. \tag{80}$$

Proof. Inserting $\xi = 0$ into Equation (79), $A(0) = a_0$ follows directly. It follows from Equation (79) that

$$\begin{aligned} &A'(\xi) = a_1 + 2\xi F(\xi) + \xi^2 F'(\xi),\ A''(\xi) = 2F(\xi) + 4\xi F'(\xi) + \xi^2 F''(\xi), \\ &A'''(\xi) = 6F'(\xi) + 6\xi F''(\xi) + \xi^2 F'''(\xi), \end{aligned}$$

of which after inserting $\xi = 0$ and using Equation (80), we can derive the last three conditions in Equation (78). □

Taking advantage of Lemma 2, we can replace Equation (76) by

$$A(\xi) = 1 - 2\xi + \xi^2 F(\xi),\ F(0) = -1,\ F'(0) = 0,\ B(0) = 1,\ B'(0) = -2,\ C(0) = 2.$$

There appear different interpolation techniques. Using a rational function, Sharma and Sharma [50] derived

$$f'(z_n) = \frac{f[x_n, z_n] f[y_n, z_n]}{f[x_n, y_n]},$$

which in terms of Equation (71) can be written as

$$f'(z_n) = \frac{h^2(\xi_n)(1-\gamma_n)(1-\eta_n)}{[h(\xi_n)+\xi_n](1-\xi_n)} f'(x_n),\ h(\xi_n) = 1 - 2\xi_n. \tag{81}$$

Based on the Taylor series expansion,

$$f'(z_n) = f[z_n, y_n] + (z_n - y_n) f[z_n, x_n, x_n] = f[z_n, y_n] + \frac{z_n - y_n}{z_n - x_n}[f[z_n, x_n] - f'(x_n)]$$

was derived by Bi et al. [34], which with the aid of Equations (71) and (72) can be recast to

$$f'(z_n) = \left[h(\xi_n)(1-\eta_n) - \frac{\xi_n[h(\xi_n)\gamma_n + \xi_n]}{[h(\xi_n)+\xi_n]^2}\right] f'(x_n),\ h(\xi_n) = \frac{2 - 5\xi_n}{2 - \xi_n}. \tag{82}$$

Both interpolations (81) and (82) are special cases of the generalized interpolation in Equation (75).

To observe the accuracy of Equation (75) and the other two interpolations (81) and (82), we consider two definite functions:

$$f_1(x) = x^3 + 4x^2 - 10,\ f_2(x) = e^{x^2+7x-30} - 1, \tag{83}$$

of which $f_1(r = 1.3652300134) = 0$ and $f_2(r = 3) = 0$. We define

$$\text{RE} := \frac{|f'(z_n) - f'_E(z_n)|}{|f'_E(z_n)|} \tag{84}$$

to be the relative error of the interpolation of $f'(z_n)$, where $f'(z_n)$ and $f'_E(z_n)$ are, respectively, the value calculated from Equation (75) and the exact value.

The following cases are with simple weight functions:

(a) $h = 1 - 2\xi_n$, and $A(\xi_n)$, $B(\xi_n)$, $C(\xi_n)$ by Equation (77),
(b) $h = 1 - 2\xi_n - \xi_n^4$, and $A(\xi_n)$, $B(\xi_n)$, $C(\xi_n)$ by Equation (74),
(c) $h = 1 - 2\xi_n + \xi_n^3$, and $A(\xi_n)$, $B(\xi_n)$, $C(\xi_n)$ by Equation (74),
(d) $h = 1 - 2\xi_n + \xi_n^4$, and $A(\xi_n)$, $B(\xi_n)$, $C(\xi_n)$ by Equation (74),
(e) $h = 1 - 2\xi_n + \xi_n^3$, $F(\xi_n) = -1 - \xi_n^2$, $B(\xi_n) = 1 - 2\xi_n$, $C(\xi_n) = 2 - 2\xi_n$,
(f) $h = 1 - 2\xi_n$, $F(\xi_n) = -1 + \xi_n^2 + \xi_n^3$, $B(\xi_n) = 1 - 2\xi_n + \xi^2$, $C(\xi_n) = 2$. (85)

Table 10 lists the REs.

Table 10. Listing REs for the functions $f_1(x)$ and $f_2(x)$ in Equation (83), which are computed at $x_n = 1.3$ and $x_n = 3.05$, respectively. Each number of $e(-k)$ means that 10^{-k}.

	(a)	**(b)**	**(c)**	**(d)**	**(e)**	**(f)**	**(81)**	**(82)**
f_1	$3.6e(-14)$	$2.4e(-14)$	$2.3e(-14)$	$1.1e(-14)$	$3.3e(-5)$	$3.8e(-6)$	$2.6e(-5)$	$2.4e(-5)$
f_2	$4.3e(-3)$	$4.3e(-3)$	$4.2e(-3)$	$4.2e(-3)$	$4.3e(-2)$	$1.6e(-2)$	$9.1e(-3)$	$2.5e(-2)$

For $f_1(x)$, the accuracy of Cases (a)–(d) is much better than others because the interpolant $f_1(x)$ is itself a third-order polynomial; however, for $f_2(x)$, the accuracy of all cases are of the levels 10^{-2} and 10^{-3}.

9. Three-Point Eighth-Order Optimal Iterative Schemes

In this section, we combine the two-point and three-point generalized Hermite interpolations to generate some eighth-order optimal iterative schemes (EOIS).

Theorem 5. *Equation (69) has eighth-order convergence, if $h(\xi_n)$ satisfies*

$$h(0) = 1,\ h'(0) = -2,\ h''(0) = 0, \tag{86}$$

and $A(\xi_n)$, $B(\xi_n)$ and $C(\xi_n)$ in

$$g(\xi_n, \eta_n, \gamma_n) = A(\xi_n) - B(\xi_n)\eta_n - C(\xi_n)\gamma_n \tag{87}$$

satisfy the conditions in Equation (76).

Proof. Before giving the proof, we emphasize that for the special case with $h(\xi_n) = 1 - 2\xi_n$ and $A(\xi_n)$, $B(\xi_n)$ and $C(\xi_n)$ given in Equation (77), Wang and Liu [36] have proven the eighth-order convergence of the iterative scheme (69) and derived the corresponding error equation. For saving space, the details are not written here, and the error equation is not written out explicitly.

Let $e_n = x_n - r$, $s_n = y_n - r$, $d_n = z_n - r$ and $c_k = f^{(k)}(r)/[k!f'(r)]$, $k = 2, \ldots$. As shown in [36],

$$f(x_n) = f'(r)[e_n + c_2e_n^2 + \ldots + c_8e_n^8 + \mathcal{O}(e_n^9)], \tag{88}$$
$$f(y_n) = f'(r)[s_n + c_2s_n^2 + c_3s_n^3 + c_4s_n^4 + \mathcal{O}(e_n^9)], \tag{89}$$
$$f(z_n) = f'(r)[d_n + c_2d_n^2 + \mathcal{O}(e_n^9)], \tag{90}$$

where

$$s_n = c_2e_n^2 - 2(c_2^2 - c_3)e_n^3 - (7c_2c_3 - 4c_2^3 - 3c_4)e_n^4 + \ldots + c_8e_n^8 + \mathcal{O}(e_n^9), \tag{91}$$
$$d_n = (c_2^3 - c_2c_3)e_n^4 + \ldots + \mathcal{O}(e_n^9). \tag{92}$$

In view of Equations (88)–(92), $(\xi_n, \eta_n, \gamma_n)$ in Equation (62) have the following asymptotic estimations:

$$\xi_n \approx \mathcal{O}(e_n),\ \eta_n \approx \mathcal{O}(e_n^2),\ \gamma_n \approx \mathcal{O}(e_n^3). \tag{93}$$

Due to

$$f'(x_n) = f'(r)[1 + 2c_2e_n + \ldots + 8c_8e_n^7 + \mathcal{O}(e_n^8)]$$

and Equation (90), the term $f(z_n)/f'(x_n)$ in Equation (69) has the following asymptotic estimation:

$$\frac{f(z_n)}{f'(x_n)} \approx \mathcal{O}(e_n^4). \tag{94}$$

Therefore, we merely need to expand g to the third order by

$$g \approx A(0) + A'(0)\xi_n + \frac{A''(0)}{2}\xi_n^2 + \frac{A'''(0)}{6}\xi_n^3 - [B(0) + B'(0)\xi_n]\eta_n - C(0)\gamma_n + \mathcal{O}(e_n^4), \quad (95)$$

where Equation (93) was taken into account.

Inserting $d_n = x_n - r$ and Equations (94) and (95) into the last one in Equation (69) yields

$$\begin{aligned} e_{n+1} &= d_n - \frac{1}{g}\frac{f(z_n)}{f'(x_n)} \\ &\approx d_n - \frac{1}{A(0) + A'(0)\xi_n + \frac{A''(0)}{2}\xi_n^2 + \frac{A'''(0)}{6}\xi_n^3 - [B(0) + B'(0)\xi_n]\eta_n - C(0)\gamma_n}\mathcal{O}(e_n^4) \approx \mathcal{O}(e_n^8), \end{aligned} \quad (96)$$

since $h(\xi_n)$ satisfies Equation (86); $A(\xi_n)$, $B(\xi_n)$ and $C(\xi_n)$ have the same values at $\xi_n = 0$ as listed in Equation (76) with those in Equation (77); and according to [36], the coefficients preceding e_n^4 to e_n^7 are zeros for the iterative scheme (69). Equation (96) indicates that Equation (69) has eighth-order convergence. Because we do not derive the error equation in an explicit form, many processes were omitted. □

Although, the details of the derivation of the corresponding error equation is not given here, we give two examples to verify the performance of the iterative scheme (69), of which $h(\xi_n)$ and $g(\xi_n, \eta_n, \gamma_n)$ are independent, not like that in [36] as shown by Case (a) in Equation (85), wherein g is related to h by Equations (75) and (74). We abbreviate the method in [36] as WLM. For the purpose of comparison, the WLM is written as follows:

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \\ z_n = y_n - \frac{f(y_n)}{2f[x_n,y_n] - f'(x_n)}, \\ x_{n+1} = z_n - \frac{f(z_n)}{2f[x_n,z_n] + f[y_n,z_n] - 2f[x_n,y_n] + (y_n - z_n)f[y_n,x_n,x_n]}, \end{cases} \quad (97)$$

$$\begin{cases} f[x_n, y_n] = \frac{f(x_n) - f(y_n)}{x_n - y_n}, \\ f[x_n, z_n] = \frac{f(x_n) - f(z_n)}{x_n - z_n}, \\ f[y_n, z_n] = \frac{f(y_n) - f(z_n)}{y_n - z_n}, \\ f[y_n, x_n, x_n] = \frac{f[x_n,y_n] - f'(x_n)}{y_n - x_n}. \end{cases} \quad (98)$$

This iterative scheme was proved in [36] to be eighth-order convergence, which is optimal, because there are four function operations on $[f(x_n), f'(x_n), f(y_n), f(z_n)]$. Compared to Equation (69), which is also eighth-order convergence, and also with the same function operations on $[f(x_n), f'(x_n), f(y_n), f(z_n)]$, the two methods are different in their construction techniques.

The iterative scheme (69), which involves $h(\xi_n)$ and $g(\xi_n, \eta_n, \gamma_n)$, is definitely more general than that in Equations (97) and (98). The theoretical basis of Equation (69) is the generalizations of the two-point and three-point Hermite interpolation methods.

Substituting the functions in Cases (a)–(f) into the iterative scheme (69), we have different algorithms; in particular, using the functions in Equations (81) and (82), we recover the iterative schemes developed in [34,50], which are shortened as SSM and BRWM in Table 11. The convergence criteria are $|x_{n+1} - x_n| < 10^{-15}$ and $|f(x_n)| < 10^{-15}$. It can be seen that these iterative schemes have the same performance.

Table 11. For the functions $f_1(x) = x^3 + 4x^2 - 10$ with $x_0 = 1$ and $f_2(x) = e^{x^2+7x-30} - 1$ with $x_0 = 3.5$, comparing the number of iterations.

	WLM	**(b)**	**(c)**	**(d)**	**(e)**	**(f)**	**SSM**	**BRWM**
$f_1 = 0$	3	3	3	3	3	3	3	3
$f_2 = 0$	6	6	6	6	6	6	6	5

10. Conclusions

We have derived a simple iterative scheme to solve nonlinear equations from a two-dimensional approach with two constant parameters involved. Executing the convergence analysis, the parameters were constrained by a derived inequality, which resulted in a third-order convergence iterative scheme. The presented method is derivative-free and the theoretically and numerically proved order of the convergence is three. We proved that the iterative scheme is a variant of Newton's method improved from quadratic convergence to cubic convergence. Numerical results revealed that the new method can be of practical interest for finding the solution quickly, and at each iteration, it merely required one evaluation of the given function. The proposed fractional iterative scheme was combined with the generalized quadrature methods to develop a class of fourth-order optimal iterative schemes. Examples were given to show that the COC is close to four. We employed a finite difference technique on the data at three points near to the solution to estimate the optimal values of two parameters, whose efficiency was identified by numerical tests. In terms of weight functions with certain constraints, two new generalized Hermite interpolation techniques were performed for the transformations between the derivatives at two points and at three points. A modification of the triple Newton's method with the two generalized Hermite interpolation formulas can realize the eighth-order optimal iterative scheme, whose E.I. = 1.682 is better than E.I. = 1.414 of the triple Newton method. Two examples were verified for testing the accuracy of the generalized Hermite interpolations and the efficiency of the proposed eighth-order optimal iterative schemes. In summary, the novelty points of this paper are as follows:

- A two-dimensional approach to the modification of the Newton method is developed.
- The resulting fractional iterative scheme involving two parameters is of the one-step type and also of third-order convergence, which saves much computation of function per iteration.
- A new fourth-order optimal iterative scheme was constructed.
- A new three-point generalized Hermite interpolation was constructed, which is quite general.
- A new eighth-order optimal iterative scheme was constructed.

Author Contributions: Methodology, C.-S.L. and C.-W.C.; Software, C.-S.L., E.R.E.-Z. and C.-W.C.; Validation, C.-S.L. and C.-W.C.; Formal analysis, C.-S.L. and C.-W.C.; Investigation, C.-S.L., E.R.E.-Z. and C.-W.C.; Resources, E.R.E.-Z. and C.-W.C.; Data curation, C.-S.L.; Writing—original draft, C.-S.L.; Writing—review & editing, C.-W.C.; Visualization, C.-S.L., E.R.E.-Z. and C.-W.C.; Supervision, C.-S.L. and C.-W.C.; Project administration, C.-W.C.; Funding acquisition and Resources, E.R.E.-Z. and C.-W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the National United University [grant number: 111I1206-8] and the National Science and Technology Council [grant number: NSTC 112-2221-E-239-022]. Also, this study is supported via funding from Prince sattam bin Abdulaziz University project number (PSAU/2023/R/1445).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BRWM:	Bi, Ren and Wu method
COC:	computed order of convergence
DFNM:	derivative-free Newton method
E.I.:	efficiency index
EOIS:	eighth-order optimal iterative scheme
FDFNM:	first derivative-free Newton method
FOIS:	fourth-order optimal iterative scheme
HM:	Halley method
LM:	Li method
NI:	number of iterations
NM:	Newton method
RE:	relative error
SDFNM:	second derivative-free Newton method
SSM:	Sharma and Sharma method
WLM:	Wang and Liu method

References

1. Chun, C.; Ham, Y. Some fourth-order modifications of Newton's method. *Appl. Math. Comput.* **2008**, *197*, 654–658. [CrossRef]
2. Weerakoon, S.; Fernando, T.G.I. A variant of Newton's method with accelerated third-order convergence. *Appl. Math. Lett.* **2000**, *13*, 87–93. [CrossRef]
3. Noor, M.A.; Noor, K.I.; Al-Said, E.; Waseem, M. Some new iterative methods for nonlinear equations. *Math. Prob. Eng.* **2000**, *2010*, 198943. [CrossRef]
4. Morlando, F. A class of two-step Newton's methods with accelerated third-order convergence. *Gen. Math. Notes* **2015**, *29*, 17–26.
5. Thukral, R. New modification of Newton method with third-order convergence for solving nonlinear equations of type $f(0) = 0$. *Am. J. Comput. Appl. Math.* **2016**, *6*, 14–18.
6. Saqib, M.; Iqbal, M. Some multi-step iterative methods for solving nonlinear equations. *Open J. Math. Sci.* **2017**, *1*, 25–33. [CrossRef]
7. Qureshi, U.K. A new accelerated third-order two-step iterative method for solving nonlinear equations. *Math. Theo. Model.* **2018**, *8*, 64–68.
8. Ali, F.; Aslam, W.; Ali, K.; Anwar, M.A.; Nadeem, A. New family of iterative methods for solving nonlinear models. *Discr. Dyn. Nat. Soc.* **2018**, *2018*, 9619680. [CrossRef]
9. Homeier, H.H.H. On Newton-type methods with cubic convergence. *J. Comput. Appl. Math.* **2005**, *176*, 425–432. [CrossRef]
10. Ozban, A.Y. Some new variants of Newton's method. *Appl. Math. Lett.* **2004**, *17*, 677–682. [CrossRef]
11. Lukic, T.; Ralevic, N.M. Geometric mean Newton's method for simple and multiple roots. *Appl. Math. Lett.* **2008**, *21*, 30–36. [CrossRef]
12. Ababneh, O.Y. New Newton's method with third-order convergence for solving nonlinear equations. *World Acad. Sci. Eng. Tech.* **2012**, *6*, 1269–1271.
13. Abdul-Hassan, N.Y. Two new predictor-corrector iterative methods with third- and ninth-order convergence for solving nonlinear equations. *Math. Theo. Model.* **2016**, *6*, 44–56.
14. Kou, J.; Li, Y.; Wang, X. Third-order modification of Newtons method. *J. Comput. Appl. Math.* **2007**, *205*, 1–5.
15. Chun, C. On the construction of iterative methods with at least cubic convergence. *Appl. Math. Comput.* **2007**, *189*, 1384–1392. [CrossRef]
16. Verma, K.L. On the centroidal mean Newton's method for simple and multiple roots of nonlinear equations. *Int. J. Comput. Sci. Math.* **2016**, *7*, 126–143. [CrossRef]
17. Liu, C.S.; Li, T.L. A new family of fourth-order optimal iterative schemes and remark on Kung and Traub's conjecture. *J. Math.* **2021**, *2021*, 5516694. [CrossRef]
18. Li, S. Fourth-order iterative method without calculating the higher derivatives for nonlinear equation. *J. Algor. Comput. Tech.* **2019**, *13*, 1–8. [CrossRef]
19. Chun, C. Some fourth-order iterative methods for solving nonlinear equations. *Appl. Math. Comput.* **2008**, *195*, 454–459. [CrossRef]
20. King, R. A family of fourth-order iterative methods for nonlinear equations. *SIAM J. Numer. Anal.* **1973**, *10*, 876–879. [CrossRef]
21. Chun, C. Certain improvements of Chebyshev-Halley methods with accelerated fourth-order convergence. *Appl. Math. Comput.* **2007**, *189*, 597–601. [CrossRef]
22. Kuo, J.; Li, Y.; Wang, X. Fourth-order iterative methods free from second derivative. *Appl. Math. Comput.* **2007**, *184*, 880–885.
23. Chun, C. Some variants of King's fourth-order family of methods for nonlinear equations. *Appl. Math. Comput.* **2007**, *190*, 57–62. [CrossRef]
24. Ostrowski, A.M. *Solutions of Equations and System Equations*; Academic Press: New York, NY, USA, 1960.

25. Maheshwari, A.K. A fourth order iterative method for solving nonlinear equations. *Appl. Math. Comput.* **2009**, *211*, 383–391. [CrossRef]
26. Ghanbari, B. A new general fourth-order family of methods for finding simple roots of nonlinear equations. *J. King Saud Univ. Sci.* **2011**, *23*, 395–398. [CrossRef]
27. Khattri, S.K.; Noot, M.A.; Al-Said, E. Unifying fourth-order family of iterative methods. *Appl. Math. Lett.* **2011**, *24*, 1295–1300. [CrossRef]
28. Kumar, S.; Kanwar, V.; Singh, S. Modified efficient families of two and three-step predictor-corrector iterative methods for solving nonlinear equations. *Appl. Math.* **2010**, *1*, 153–158. [CrossRef]
29. Chicharro, F.I.; Cordero, A.; Garrido, N.; Torregrosa, J.R. Wide stability in a new family of optimal fourth-order iterative methods. *Comput. Math. Meth.* **2019**, *1*, e1023. [CrossRef]
30. Liu, D.; Liu, C.S. Two-point generalized Hermite interpolation: Double-weight function and functional recursion methods for solving nonlinear equations. *Math. Comput. Simul.* **2022**, *193*, 317–330. [CrossRef]
31. Cordero, A.; Hueso, J.L.; Martinez, E.; Torregrosa, J.R. New modifications of Potra-Ptak's method with optimal fourth and eighth orders of convergence. *J. Comput. Appl. Math.* **2010**, *234*, 2969–2976. [CrossRef]
32. Chun, C.; Ham, Y. Some sixth-order variants of Ostrowski root-finding methods. *Appl. Math. Comput.* **2007**, *193*, 389–394. [CrossRef]
33. Kou, J.; Li, Y.; Wang, X. Some variants of Ostrowskis method with seventh-order convergence. *J. Comput. Appl. Math.* **2007**, *209*, 153–159. [CrossRef]
34. Bi, W.; Ren, H.; Wu, Q. Three-step iterative methods with eighth-order convergence for solving nonlinear equations. *J. Comput. Appl. Math.* **2009**, *225*, 105–112. [CrossRef]
35. Bi, W.; Wu, Q.; Ren, H. A new family of eighth-order iterative methods for solving nonlinear equations. *Appl. Math. Comput.* **2009**, *214*, 236–245. [CrossRef]
36. Wang, X.; Liu, L. Modified Ostrowski's method with eighth-order convergence and high efficiency index. *Appl. Math. Lett.* **2010**, *23*, 549–554. [CrossRef]
37. Petković, M.S. On optimal multipoint methods for solving nonlinear equations. *Novi Sad J. Math.* **2009**, *39*, 123–130.
38. Petković, M.S.; Petković, L.D. Families of optimal multipoint methods for solving nonlinear equations: A survey. *Appl. Anal. Discr. Math.* **2010**, *4*, 1–22.
39. Neta, B.; Petković, M.S. Construction of optimal order nonlinear solvers using inverse interpolation. *Appl. Math. Comput.* **2010**, *217*, 2448–2455. [CrossRef]
40. Soleymani, F.; Shateyi, S.; Salmani, H. Computing simple roots by an optimal sixteenth-order class. *J. Appl. Math.* **2012**, *2012*, 958020. [CrossRef]
41. Matinfar, M.; Aminzadeh, M. Three-step iterative methods with eighth-order convergence for solving nonlinear equations. *J. Interpol. Approx. Sci. Comput.* **2013**, *2013*, jiasc-00013. [CrossRef]
42. Zafar, F.; Yasmin, N.; Akram, S.; Junjua, M.D. A general class of derivative-free optimal root finding methods based on rational interpolation. *Sci. World J.* **2015**, *2015*, 935260. [CrossRef] [PubMed]
43. Junjua, M.D.; Zafar, F.; Yasmin, N. Optimal derivative-free root finding methods based on inverse interpolation. *Mathematics* **2019**, *7*, 164. [CrossRef]
44. Zhanlav, T.; Chuluunbaatar, O.; Ulziibayar, V. Generating function method for constructing new iterations. *Appl. Math. Comput.* **2017**, *315*, 414–423. [CrossRef]
45. Liu, C.S. A new splitting technique for solving nonlinear equations by an iterative scheme. *J. Math. Res.* **2020**, *12*, 40–48. [CrossRef]
46. Liu, C.S.; Hong, H.K.; Lee, T.L. A splitting method to solve a single nonlinear equation with derivative-free iterative schemes. *Math. Comput. Simul.* **2021**, *190*, 837–847. [CrossRef]
47. Halley, E. A new exact and easy method for finding the roots of equations generally and without any previous reduction. *Philos. Roy. Soc. London* **1964**, *8*, 136–147.
48. Liu, C.S.; Li, T.L. A new family of generalized quadrature methods for solving nonlinear equations. *Asian-Eur. J. Math.* **2022**, *15*, 2250044. [CrossRef]
49. Cordero, A.; Franceschi, J.; Torregrosa, J.R.; Zagati, A.C. A convex combination approach for mean-based variants of Newton's method. *Symmetry* **2019**, *11*, 1106. [CrossRef]
50. Sharma, J.R.; Sharma, R. A new family of modified Ostrowski's methods with accelerated eighth order convergence. *Numer. Algorithms* **2010**, *54*, 445–458. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.