mathematics

MDPI

Article

A New Family of Methods for Solving Systems of Nonlinear Equations

Đorđe Herceg [1], Dejana Herceg [2], Péter Odry [3] and Vladimir Tadić [4,5,*]

[1] Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, 21000 Novi Sad, Serbia; herceg@dmi.uns.ac.rs
[2] Department of Power, Electronics and Telecommunication Engineering, Faculty of Technical Sciences, University of Novi Sad, 21000 Novi Sad, Serbia; vuletic@uns.ac.rs
[3] Institute of Information Technology, University of Dunaujvaros, Tancsics M. Str. 1/A, H-2401 Dunaujvaros, Hungary; podry@uniduna.hu
[4] Department of Mechanical Engineering, Electrical Engineering and Computer Science, Technical College of Applied Sciences in Zrenjanin, Đorđa Stratimirovića 23, 23000 Zrenjanin, Serbia
[5] John von Neumann Faculty of Informatics, Obuda University, Becsi Str. 96/B, H-1034 Budapest, Hungary
* Correspondence: vladimir.tadic@vts-zr.edu.rs

Abstract

A new family of iterative methods for solving systems of nonlinear equations is introduced, extending and unifying several existing approaches within a general framework. In the single-variable case, Newton's method was accelerated using Padé approximations. This strategy is then extended to systems of nonlinear equations, leading to the construction of higher-order iteration functions with convergence orders ranging from two to four. By varying the order (p, q) of the Padé approximation, several known methods and their modifications emerge as special cases of this generalized approach. Theoretical convergence rates are established, and numerical experiments, conducted using high-precision arithmetic, confirm both the predicted behavior and improved computational efficiency across a variety of test problems.

Keywords: systems of nonlinear equations; Newton's method; higher order methods; iterative methods

MSC: 65H05

check for updates

Academic Editor: Alicia Cordero

Received: 5 November 2025
Revised: 28 November 2025
Accepted: 9 December 2025
Published: 11 December 2025

Citation: Herceg, Đ.; Herceg, D.; Odry, P.; Tadić, V. A New Family of Methods for Solving Systems of Nonlinear Equations. *Mathematics* **2025**, *13*, 3945. https://doi.org/10.3390/math13243945

Copyright: © 2025 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

A new family of iterative methods for numerical solution of systems of nonlinear equations, achieving orders of convergence from two to four, is developed and presented. The main contribution of this work is in the generalization and extension of several well-known approaches, including methods originally formulated for single-variable nonlinear equations [1].

Methods that accelerate Newton's method are commonly encountered in the literature. Multistep methods of order three, four, and higher can be constructed using weight functions [1–4]. The convergence order of classical methods has been improved through numerous approaches, including those of Chebyshev, Halley, King, Newton–Steffensen, Ostrowski, and Potra-Pták; see [1] and references therein.

Methods that avoid second- and higher-order derivatives are of particular interest for nonlinear equations [1,5–9]. Likewise, approaches that optimize or eliminate function, matrix, and Jacobian evaluations, while emphasizing computational efficiency, are

especially relevant in the context of nonlinear systems [10–16]. For example, in [15], the authors propose a parametric fourth-order method that reduces evaluation complexity by employing divided differences in place of full Jacobian evaluations. In [17], a parametric family of fourth-order Chebyshev–Halley type methods for systems of nonlinear equations is introduced. Fourth-order convergence is achieved in just two steps, without requiring second derivatives. In [18], a one-parameter family of iterative methods for solving nonlinear systems is proposed that avoids Jacobian computation by using divided differences. Fourth-order convergence is attained with minimal computational cost, requiring only one function evaluation, one divided difference, and one matrix inversion per iteration. Emphasis is placed on maximizing computational efficiency and balancing convergence order with operation count. In [19], a new class of frozen Jacobian multi-step iterative methods is presented, requiring only one Jacobian evaluation and LU decomposition per iteration. The two-step method achieves third-order convergence, while the three-step method is of order four. In [20], a new fourth-order parametric family is introduced. This two-step iterative method achieves fourth-order convergence using a single Jacobian inversion per iteration. The method depends on one free parameter, allowing for tuning in terms of stability and performance. The concept of maximally efficient iterative schemes is introduced in [21], defined as methods that attain the highest possible efficiency index for a given convergence order. The work builds on the Cordero–Torregrosa conjecture, which generalizes Kung–Traub's optimality principle to vector-valued systems and helps define new metrics for method comparison. The approach is validated on large-scale nonlinear systems.

A central feature of this work is the use of Padé approximations to construct higher-order iteration functions. The well-known Cauchy's method is defined as

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

where

$$L_f(x_n) = \frac{f''(x_n)f(x_n)}{f^2(x_n)}.$$

However, the method depends on the square root function and second derivatives that can make the iterative process costly or impractical. To address this, the term $2/\left(1+\sqrt{1-2z}\right)$ is replaced with a rational approximant, and a suitable substitute for $L_f(x_n)$ is chosen. Padé approximation of order (p, q) is employed as the rational approximant. By varying the order (p, q) of the Padé approximation, one obtains a family of methods, including extensions of several classical root-finding methods to the setting of systems of nonlinear equations.

We establish the theoretical convergence rates of the proposed methods and validate them through high-precision numerical experiments, demonstrating both accuracy and computational efficiency.

The paper is structured as follows: Section 2 presents the development of the new family of methods, based on the generalization of single-variable techniques to nonlinear systems. Section 3 provides the convergence analysis. Section 4 illustrates several well-known methods that fall within our proposed family. Section 5 offers numerical examples, along with a discussion and comparison of computational complexity between the proposed methods and other recently developed approaches.

Table 1 summarizes the third- and fourth-order methods cited in references, including their convergence order and computational complexity, expressed in terms of function evaluations, first derivative evaluations, and matrix inversions or linear system solutions performed in each iterative step.

Table 1. Comparison of methods by the number of function evaluations and matrix inversions.

References	Convergence Order	f Evaluations	f' or Divided Difference Evaluations	Matrix Inversions or Linear System Solutions
[17]	4	1	2	2
[18]	4	2	2	2
[19]	4	3	1	3
[20]	4	1	2	2
[21]	4	2	1	1
[22–30]	3	1	2	2
[31]	3	1	3	2
[31]	3	1	4	2
[32]	3	2	1	1
[33]	4	2	2	1
[34]	4	1	2	2
[35]	4	1	2	2
[36]	4	1	2	2
[26]	4	3	1	1
[37]	4	1	2	2
[38]	4	2	3	2

2. Development of the Methods

We consider a family of iterative methods for solving systems of nonlinear equations $f(x) = 0$, where $f : R^n \to R^n$ is assumed to be continuously differentiable on an open convex set $D \subset R^n$. We also assume that there exists $\alpha \in D$ such that $f(\alpha) = 0$, and that the Fréchet derivative $f'(x)$ is continuous and nonsingular in α.

Let S be an $n \times n$ matrix, and let $\varphi_{p,q}$ be defined as the Padé approximation of order (p, q) to the function $2/\left(1 + \sqrt{1 - 2z}\right)$ at 0. Padé approximations for $p, q = 0, 1, 2, 3$ are shown in Table 2.

Table 2. Padé approximations for $p, q = 0, 1, 2, 3$.

$p \backslash q$	0	1	2	3
0	1	$\frac{-2}{z-2}$	$\frac{-4}{z^2+2z-4}$	$\frac{-4}{z^3+z^2+2z-4}$
1	$\frac{z}{2}+1$	$\frac{z-2}{2(z-1)}$	$\frac{4-4z}{z^2-6z+4}$	$\frac{16-20z}{z^3+6z^2-28z+16}$
2	$\frac{z^2}{2}+\frac{z}{2}+1$	$\frac{z^2+6z-8}{10z-8}$	$\frac{z^2-6z+4}{3z^2-8z+4}$	$\frac{-2\left(3z^2-8z+4\right)}{z^3-12z^2+20z-8}$
3	$\frac{5z^3}{8}+\frac{z^2}{2}+\frac{z}{2}+1$	$\frac{3z^3+8z^2+36z-40}{8(7z-5)}$	$\frac{z^3+12z^2-44z+24}{28z^2-56z+24}$	$\frac{z^3-12z^2+20z-8}{4(z^3-5z^2+6z-2)}$

The matrices $A_{p,q}$ and $B_{p,q}$ are defined as follows. Let polynomials g and h denote the numerator and the denominator of the function $\varphi_{p,q}$:

$$g(z) = g_0 + g_1 z + g_2 z^2 + \ldots + g_k z^k \tag{1}$$

$$h(z) = h_0 + h_1 z + h_2 z^2 + \ldots + h_m z^m \tag{2}$$

By evaluating these polynomials at the matrix S, we obtain the matrices

$$A_{p,q}(S) = g_0 E + g_1 S + g_2 S^2 + \ldots + g_k S^k \tag{3}$$

$$B_{p,q}(S) = h_0 E + h_1 S + h_2 S^2 + \ldots + h_k S^m \tag{4}$$

where E denotes the $n \times n$ identity matrix. Based on $A_{p,q}(S)$ and $B_{p,q}(S)$, we define the matrix as

$$\Phi_{p,q}(S) = B_{p,q}(S)^{-1} A_{p,q}(S) \tag{5}$$

This way, the Padé approximation $\varphi_{p,q}$ is associated with the matrix $\Phi_{p,q}$.

From now on, we consider a family of iterative methods defined by

$$x^{(k+1)} = F_{p,q}\left(x^{(k)}\right), \quad k = 0, 1, \ldots \tag{6}$$

with the iteration function $F_{p,q}$ of the form

$$F_{p,q}(x) = x - \Phi_{p,q}(\Lambda(x))\left(f'(x)\right)^{-1} f(x), \quad p, q = 0, 1, \ldots \tag{7}$$

where

$$\Lambda(x) = \frac{3}{2}\left(E - \left(f'(x)\right)^{-1} f'\left(x - \frac{2}{3}\left(f'(x)\right)^{-1} f(x)\right)\right) \tag{8}$$

The proposed family (PM) can also be written as

$$y^{(k)} = x^{(k)} - \frac{2}{3}\left(f'\left(x^{(k)}\right)\right)^{-1} f\left(x^{(k)}\right) \tag{9}$$

$$\Lambda\left(x^{(k)}\right) = \frac{3}{2}\left(E - \left(f'\left(x^{(k)}\right)\right)^{-1} f'\left(y^{(k)}\right)\right) \tag{10}$$

$$x^{(k+1)} = x^{(k)} - \Phi_{p,q}\left(\Lambda\left(x^{(k)}\right)\right)\left(f'\left(x^{(k)}\right)\right)^{-1} f\left(x^{(k)}\right). \tag{11}$$

Theorem 1, which establishes the convergence of the proposed family of methods, is stated and proved in the following section.

3. Convergence Analysis

Theorem 1. *Assume the vector function $f : D \subset R^n \to R^n$ be k-time Fréchet differentiable in a convex set D containing a solution α of $f(x) = 0$ and the Fréchet derivative $f'(x)$ is continuous and nonsingular in α. Then the family of methods (9), (10), and (11) has an order of convergence $ord_{p,q} = \min\{p + q + 2, 4\}$. The corresponding asymptotic error constants are $E_{p,q} = \delta$ for $p + q > 2$, and in other cases, as in Table 3, with*

$$C_j = (1/j!)\left(f'(\alpha)\right)^{-1} f^{(j)}(\alpha), \quad j = 1, 2, \ldots \tag{12}$$

and

$$\delta = -C_3 C_2 + \frac{1}{9} C_4 \tag{13}$$

Table 3. Coefficients $E_{p,q}$ for $p + q \leq 2$.

$p \backslash q$	0	1	2
0	C_2	C_2^2	$2C_2^3 + \delta$
1	$2C_2^2$	$C_2^3 + \delta$	
2	$5C_2^3 + \delta$		

Proof of Theorem 1. Let us consider the proposed family of methods (9), (10), and (11). For the sake of simplicity, let us omit the indices p, q by A and B, and let us denote $e = x - \alpha$. Now, with C_j given by (12), we have

$$f(x) = f'(\alpha)\left(e + C_2e^2 + C_3e^3 + C_4e^4\right) + O\left(e^5\right)$$
$$f'(x) = f'(\alpha)\left(E + 2C_2e + 3C_3e^2 + 4C_4e^3 + 5C_5e^4\right) + O\left(e^5\right)$$

$$\left(f'(x)\right)^{-1} = \left(E - 2C_2e + \left(4C_2^2 - 3C_3\right)e^2 + R_3e^3\right)\left(f'(\alpha)\right)^{-1} + O\left(e^4\right) \tag{14}$$

where

$$R_3 = -8C_2^3 + 6C_2C_3 + 6C_3C_2 - 4C_4$$

It directly follows that

$$\left(f'(x)\right)^{-1}f(x) = \left(e - C_2e^2 + \left(2C_2^2 - 2C_3\right)e^3 + \left(-4C_2^3 + 4C_2C_3 + 3C_3C_2 - 3C_4\right)e^4\right) + O\left(e^5\right) \tag{15}$$

and the following relations are obtained:

$$y = x - \tfrac{2}{3}\left(e - C_2e^2 + \left(2C_2^2 - 2C_3\right)e^3 + \left(-4C_2^3 + 4C_2C_3 + 3C_3C_2 - 3C_4\right)e^4\right) + O\left(e^5\right)$$
$$y - \alpha = \tfrac{1}{3}\left(e + 2C_2e^2 - 4\left(C_2^2 - C_3\right)e^3 - 2\left(-4C_2^3 + 4C_2C_3 + 3C_3C_2 - 3C_4\right)e^4\right) + O\left(e^5\right)$$
$$(y - \alpha)^2 = \tfrac{1}{9}\left(e^2 + 4C_2e^3 - 4\left(C_2^2 - 2C_3\right)\right)e^4 + O\left(e^5\right)$$
$$(y - \alpha)^3 = \tfrac{1}{27}\left(e^3 + 6C_2e^4\right) + O\left(e^5\right)$$
$$(y - \alpha)^4 = \tfrac{1}{81}e^4 + O\left(e^5\right).$$

Since

$$f'(y) = f'(\alpha)\left(E + 2C_2(y - \alpha) + 3C_3(y - \alpha)^2 + 4C_4(y - \alpha)^3\right) + O\left((y - \alpha)^4\right), \tag{16}$$

from (14) and (16), the following is derived:

$$\left(f'(x)\right)^{-1}f'(y) = \left(E - 2C_2e + \left(4C_2^2 - 3C_3\right)e^2 + \left(-8C_2^3 + 6C_2C_3 + 6C_3C_2 + 4C_4\right)e^3\right)$$
$$\left(E + 2C_2(y - \alpha) + 3C_3(y - \alpha)^2 + 4C_4(y - \alpha)^3\right) + O\left((y - \alpha)^4\right) + O\left(e^5\right)$$

$$\left(f'(x)\right)^{-1}f'(y) = \frac{2}{3}\left\{E - 2C_2e + \left(6C_2^2 - 4C_3\right)e^2 + W_3e^3\right\} + O\left(e^4\right)$$

where

$$W_3 = -16C_2^3 - \frac{52C_4}{9} + 8C_2C_3 + 12C_3C_2$$

This leads to the following:

$$\begin{aligned}
\Lambda(x) &= \tfrac{3}{2}\left(E - \left(f'(x)\right)^{-1}f'(y)\right) \\
&= 2C_2e + \left(4C_3 - 6C_2^2\right)e^2 + \left(16C_2^3 + \tfrac{52C_4}{9} - 8C_2C_3 - 12C_3C_2\right)e^3 + O\left(e^4\right) \\
\Lambda(x)^2 &= 4C_2^2e^2 + \left(-24C_2^3 + 8C_3C_2 + 8C_2C_3\right)e^3 + O\left(e^4\right) \\
\Lambda(x)^3 &= 8C_2^3e^3 + O\left(e^4\right)
\end{aligned} \tag{17}$$

The Padé approximation of order (p, q) is a rational function whose Taylor expansion agrees with the ordinary series expansion up to order $p + q$. Different choices of the parameters p and q yield different rational approximants. For our discussion, the following approximation for $\Phi_{p,q}(\Lambda(x)) = B(\Lambda(x))^{-1}A(\Lambda(x))$ is of interest:

$$\Phi_{p,q}(\Lambda(x)) = E + \frac{a}{2}\Lambda(x) + \frac{b}{2}\Lambda(x)^2 + \frac{d}{8}\Lambda(x)^3 + O\left(e^4\right) \tag{18}$$

with different values of a, b and d depending on $p + q$. The form (18) is chosen to facilitate a simpler and more transparent analysis.

Let $e = x^{(k)} - \alpha$ and $\hat{e} = x^{(k+1)} - \alpha$. From (15) and (18), we obtain the following by direct multiplication:

$$\begin{aligned}\hat{e} = e - \Phi_{p,q}(\Lambda(x))(f'(x))^{-1} f(x) = (1-a)C_2e^2 + 2\big((2a-b-1)C_2^2 + (1-a)C_3\big)e^3 + \\ + \big((-13a+14b-d+4)C_2^3 + 4(2a-b-1)C_2C_3 + (6a-4b-3)C_3C_2 + \big(3 - \tfrac{26a}{9}\big)C_4\big)e^4 + O\big(e^5\big)\end{aligned} \tag{19}$$

Two cases are distinguished. In the first case, when $p+q \leq 2$, the corresponding values of a, b, and d are listed in Table 4. In the second case, for $p+q > 2$, the parameters take the fixed values $a = b = 1$ and $d = 5$.

Table 4. Coefficients (a, b, d) associated with (18).

$p \backslash q$	0	1	2
0	(0, 0, 0)	(1, 1/2, 1)	(1, 1, 3)
1	(1, 0, 0)	(1, 1, 4)	
2	(1, 1, 0)		

For each choice of p, q, the following expressions are obtained from (19):

$$\hat{e} = e - \Phi_{p,q}(\Lambda(x))\big(f'(x)\big)^{-1} f(x) = \begin{cases} C_2 + O\big(e^3\big), & p=0,\ q=0 \\ C_2^2 + O\big(e^4\big), & p=0,\ q=1 \\ 2C_2^3 + \frac{1}{9}C_4 - C_3C_2 + O\big(e^5\big), & p=0,\ q=2 \\ 2C_2^2 + O\big(e^4\big), & p=1,\ q=0 \\ C_2^3 + \frac{1}{9}C_4 - C_3C_2 + O\big(e^5\big), & p=1,\ q=1 \\ 5C_2^3 + \frac{1}{9}C_4 - C_3C_2 + O\big(e^5\big), & p=2,\ q=0 \\ \frac{1}{9}C_4 - C_3C_2 + O\big(e^5\big) & p+q>2 \end{cases}$$

With δ from (13), we obtain asymptotic error constants $E_{p,q}$ from Table 3.

The order of convergence of the presented family of methods is four for $p+q > 1$. In the remaining three cases, for $p+q = 0$ (Newton's method) the order of convergence is two, and for $p+q = 1$ the order of convergence is three. □

4. Some Special Cases of the Family

Some well-known methods for solving nonlinear equations are special cases of our family for simple roots [1]. Their extensions to systems of nonlinear equations are likewise special cases of our family of methods for such systems. This can be illustrated on three fourth-order methods, the first from [39], the second from [35,39], and the third being Jarratt's method.

4.1. First Method

The general form of the first method [39] is given by

$$x^{(k+1)} = G_1\left(x^{(k)}\right) \tag{20}$$

$$\begin{gathered} y(x) = x - \tfrac{2}{3}(f'(x))^{-1} f(x),\ \tau(x) = (f'(x))^{-1} f'(y(x)), \\ G_1(x) = x - H_1(x)(f'(x))^{-1} f(x), \\ H_1(x) = E + \tfrac{3}{4}(\tau(x) - E) + \tfrac{9}{8}(\tau(x) - E)^2. \end{gathered}$$

Using $\tau(x) = E - 2\Lambda(x)/3$, we directly obtain

$$H_1(x) = \Phi_{2,0}(\Lambda(x)) = E + \frac{1}{2}\Lambda(x) + \frac{1}{2}\Lambda(x)^2$$

The method (20) belongs to the proposed family (6) for $p = 2$ and $q = 0$. According to Theorem 1, the order of convergence is $ord = \min\{p + q + 2, 4\} = 4$.

4.2. Second Method

The general form of the second method [35,39] is given by

$$x^{(k+1)} = G_2\left(x^{(k)}\right) \tag{21}$$

$$\begin{gathered} G_2(x) = x - W(x)(A_1(x))^{-1} f(x) \\ A_1(x) = \tfrac{1}{2}(f'(x) + f'(y(x))) = \tfrac{1}{2} f'(x)(E + \tau(x)) \\ W(x) = E - \tfrac{1}{4}(\tau(x) - E) + \tfrac{3}{4}(\tau(x) - E)^2 \end{gathered}$$

Using $\tau(x) = E - 2\Lambda(x)/3$, we obtain

$$\begin{aligned} W(x)(A_1(x))^{-1} &= (6E - 2\Lambda(x))^{-1}\left(2\Lambda(x)^2 + \Lambda(x) + 6E\right) \\ &= E + \tfrac{1}{2}\Lambda(x) + \tfrac{1}{2}\Lambda(x)^2 + O\left(\Lambda(x)^3\right) \\ W(x)(A_1(x))^{-1} &\approx \Phi_{2,0}(\Lambda(x)) \end{aligned}$$

The method (21) belongs to our family (6) for $p = 2$ and $q = 0$. According to Theorem 1, the order of convergence is $ord = \min\{p + q + 2, 4\} = 4$.

4.3. Jarratt's Method

The general form of Jarratt's method [2] is

$$x^{(k+1)} = G_{4J}\left(x^{(k)}\right), \tag{22}$$

$$\begin{gathered} G_{4J}(x) = x - H_3(x)(f'(x))^{-1} f(x), \\ H_3(x) = (6f'(y(x)) - 2f'(x))(3f'(y(x)) + f'(x)). \end{gathered}$$

Using $\tau(x) = E - 2\Lambda(x)/3$, we obtain

$$H_3(x) = (6\tau(x) - 2E)^{-1}(3\tau(x) + E) = (2(\Lambda(x) - E))^{-1}(\Lambda(x) - 2E) = \Phi_{1,1}(\Lambda(x))$$

The method (22) belongs to our family (6) for $p = 1$ and $q = 1$. According to Theorem 1, the order of convergence is $ord = \min\{p + q + 2, 4\} = 4$.

5. Numerical Examples

All computations were carried out using high-precision arithmetic with up to 20,000 digits in Mathematica 8. For each group of nonlinear systems, we used the exit criterion, which was used in the respective original paper. Initial values were also taken from the corresponding original papers for each system. They are listed either with the systems or in tables.

The proposed family of methods is denoted by PM. In all cases, the computational order of convergence of our methods was very close to the corresponding theoretical order of convergence, which supports the theoretical result obtained in this paper. Theoretical order of convergence given by $ord_{p,q} = min\{p + q + 2, 4\}$. Members of the proposed family with third-order convergence are denoted by PM(3), and those with fourth-order convergence by PM(4).

Every iterate $x^{(k+1)}$ is obtained from the previous one, $x^{(k)}$, by adding one or more terms of the form $M^{-1}v$ where M is a real $n \times n$ matrix and $v \in R^n$. The matrix M and the vector v differ across the methods used, but in any case, the inverse calculation $M^{-1}v$

is carried out, solving the linear system $Mu = v$ using Gaussian elimination with partial pivoting. This part of the calculation can be delegated to standard linear solvers, such as the LinearSolve function in Mathematica.

Let

$$x = x^{(k)},\ \hat{x} = x^{(k+1)},\ \text{and}\ y = x - \frac{2}{3}\left(f'(x)\right)^{-1} f(x).$$

Using

$$\begin{gathered} \mathrm{LM} = \mathrm{LinearSolve}\,[f'(x)], \\ y = x - \tfrac{2}{3}\mathrm{LM}[f(x)],\ \text{and} \\ \left(f'(x)\right)^{-1} f(x) = \tfrac{3}{2}(x - y) \end{gathered}$$

we obtain

$$\Lambda(x) = \frac{3}{2}\mathrm{LM}\left[f'(x) - f'(y)\right].$$

Since it holds that

$$f'(x)\Lambda(x) = \frac{3}{2}\left(f'(x) - f'\left(x - \frac{2}{3}\left(f'(x)\right)^{-1} f(x)\right)\right) = \frac{3}{2}\left(f'(x) - f'(y)\right),$$

from (9), (10), and (11), it follows that

$$\hat{x} - x = -B_{p,q}(\Lambda(x))^{-1} A_{p,q}(\Lambda(x))\left(f'(x)\right)^{-1} f(x) = -\frac{3}{2} B_{p,q}(\Lambda(x))^{-1} A_{p,q}(\Lambda(x))(x - y)$$

and

$$\begin{gathered} B_{p,q}(\Lambda(x))(\hat{x} - x) = -\tfrac{3}{2} A_{p,q}(\Lambda(x))(x - y) \\ \mathrm{FB} = \mathrm{LinearSolve}\left[B_{p,q}(\Lambda(x))\right] \\ \hat{x} = x - \mathrm{FB}\left[\tfrac{3}{2} A_{p,q}[\Lambda(x)](x - y)\right] \end{gathered}$$

Within the procedure, each iteration involves two Jacobian evaluations and one function evaluation, accompanied by two LU factorizations used to construct the LM and FB operators, for PM members with fourth-order convergence. Subsequently, three linear systems are solved—two using the LM operator and one using the FB operator. These computational steps are comparable to the profiles presented in Table 1.

For $q = 0$ and any p, the FB operator is omitted, resulting in only two linear system solves per iteration. Nevertheless, for $p > 1$, the method retains fourth-order convergence, consistent with the theoretical results.

The computational cost of the PM family can be compared with the fourth-order method (2) from [17], which will be referred to as KC. Both the PM family and the KC method share the same structure, as both involve one function evaluation, two Jacobian evaluations, and the solution of two linear systems. The first step is identical in both methods. The distinction appears in the second step, where the involved matrices are different. In the general case of the KC method, matrices are raised to the second or third power, except in the special case for $a = 1$, where no matrix powers are needed. In PM, the maximum matrix exponent is limited to $\max(p, q) - 1$. Additionally, for $q = 0$, the PM family requires one fewer linear system solution. Notably, both KC and PM include Jarratt's method as a special case.

Following the analysis outlined in [18], the total number of products and quotients required by fourth-order PM family methods can be expressed as follows:

$$op_{PM4} = \frac{5n^3}{3} + (1 + \max(p, q))n^2 - \frac{2n}{3},$$

while the fourth-order KC methods require a similar number of operations, depending on the value of the parameter a:

$$op_{KC} = \begin{cases} \frac{5n^3}{3} + 2n^2 - \frac{2n}{3}, & \text{for } a = 1, \\ \frac{5n^3}{3} + 5n^2 - \frac{2n}{3}, & \text{for } a \neq 1. \end{cases}$$

In Figure 1, the operational efficiency index $I_O = p^{1/op}$, as defined in [18], is compared across different parameter values for the PM(4) and KC methods. It can be concluded that the computational cost of our proposed family lies within the range of the computational costs of the KC family.

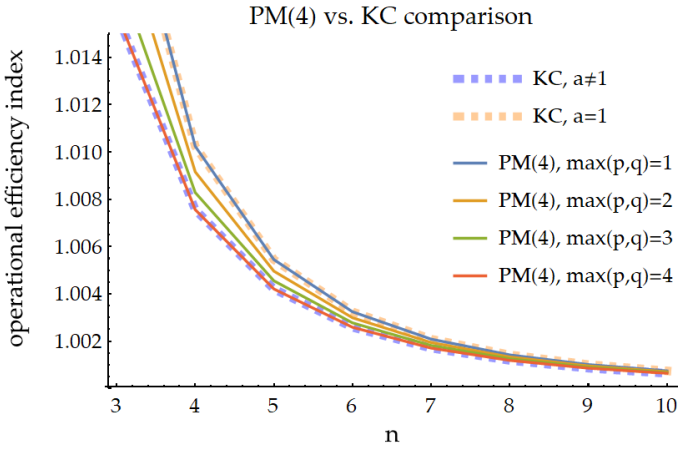


Figure 1. Operational efficiency index comparison between PM(4) and KC methods.

In the presented examples, the equation $f(x) = 0$ was solved using test functions from [10–12,17–20,31,33,35,36,40–42] with corresponding starting values.

Example 1. *We consider the following nonlinear systems of equations: system (1f), appearing in [10,12,20,36], system (1g), also appearing in [42], and the remaining systems, taken from [33]:*

(1a) $f(x) = \left\{\exp\left({x_1}^2\right) - \exp\left(\sqrt{2}x_1\right), x_1 - x_2\right\}, \alpha^{(1)} = \left\{\sqrt{2}, \sqrt{2}\right\}, \alpha^{(2)} = \{0, 0\}$

(1b) $f(x) = \{x_1 + \exp(x_2) - \cos(x_2), 3x_1 - x_2 - \sin(x_2)\}, \alpha = \{0, 0\}$

(1c) $f(x) = \left\{{x_1}^2 - 2x_1 - x_2 + 0.5, {x_1}^2 + 4{x_2}^2 - 4\right\}, \alpha = \{1.900676726\ldots, 0.311218565\ldots\}$

(1d) $f(x) = \left\{{x_1}^2 + {x_2}^2 - 1, {x_1}^2 - {x_2}^2 + 0.5\right\}, \alpha^{(1)} = \left\{1/2, \sqrt{3}/2\right\}, \alpha^{(2)} = -\alpha^{(1)}$

(1e) $f(x) = \{\sin(x_1) + x_2\cos(x_1), x_1 - x_2\}, \alpha = \{0, 0\}$

(1f) $f(x) = \{x_i x_{i+1} = 1, \ i = 1, \ldots, n-1\} \cup \{x_n x_1 = 1\}$, for odd n, $n = 99, \ n = 199, \ n = 399$, $\alpha^{(1)} = \{1, 1, \ldots, 1\}, \alpha^{(2)} = \{-1, -1, \ldots, -1\}$

(1g) $f(x) = \{x_2x_3 + x_4(x_2 + x_3), x_1x_3 + x_4(x_1 + x_3), x_1x_2 + x_4(x_1 + x_2), x_1x_2 + x_1x_3 + x_2x_3 - 1\}$, $\alpha^{(1)} = \left\{-\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, \frac{1}{2\sqrt{3}}\right\}, \alpha^{(2)} = \left\{\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, -\frac{1}{2\sqrt{3}}\right\}$

Exact solutions are denoted as α or, if there are two exact solutions, as $\alpha^{(1)}$ and $\alpha^{(2)}$. In cases where the exact solution was not available, we used approximations.

The computational order of convergence (COC) was calculated as

$$COC \approx \frac{\ln\left(\left\|x^{(k+2)} - \alpha\right\| / \left\|x^{(k+1)} - \alpha\right\|\right)}{\ln\left(\left\|x^{(k+1)} - \alpha\right\| / \left\|x^{(k)} - \alpha\right\|\right)}.$$

In the presented examples, a COC value of 4 was reached after several initial iterations.

Corresponding starting values $x^{(0)}$ for the following systems of equations are listed in Table 5. The stopping criterion, as in [33], was

$$\left\| f\left(x^{(k)}\right) \right\|_\infty + \left\| x^{(k)} - x^{(k-1)} \right\|_\infty < 10^{-12}$$

Table 5. Comparison of iteration counts for PM(3), PM(4), and reference methods in Example 1.

$f(x)$		$x^{(0)}$	CN(2)	Tr(3)	NAd(4)	PM(3)		PM(4)	
			Iter	*Iter*	*Iter*	(p, q)	*Iter*	(p, q)	*iter*
(1a)	1:	$\{2.3, 2.3\}^T$	10	8	7	(0, 1)	7	(2, 1), (4, 4)	4
	2:	$\{1.8, 1.8\}^T$	7	5	5	(0, 1)	5	(4, 0), (3, 1), (1, 1), (2, 1), (1, 2), (0, 3), (0, 4)	4
	3:	$\{0.8, 0.8\}^T$	5	4	3	(0, 1)	4	all	4
(1b)	1:	$\{1.5, 2\}^T$	7	6	5	(0, 1)	5	(3, 0), (4, 0), (1, 1), (2, 1), (1, 2), (0, 3), (0, 4)	4
	2:	$\{0.3, 0.5\}^T$	5	4	4	(1, 0), (0, 1)	4	(2, 1), (3, 1), (4, 1), (1, 2), (2, 2), (3, 2), (4, 2), (1, 3), (2, 3), (3, 3), (4, 3), (0, 4), (1, 4), (2, 4); (3, 4), (4, 4)	3
(1c)	1:	$\{3, 2\}^T$	7	5	5	(1, 0), (0, 1)	5	all except (2, 0)	4
	2:	$\{1.6, 0\}^T$	5	4	4	(0, 1)	4	(1, 2)	3
(1d)	1:	$\{0.7, 1.2\}^T$	5	4	3	(1, 0), (0, 1)	4	all except (2, 0), (0, 2)	3
	2:	$\{-1, -2\}^T$	6	4	4	(0, 1), (1, 0)	5	(3, 1), (4, 1), (2, 2), (3, 2), (4, 2), (1,3), (2, 3), (3, 3), (4, 3), (1, 4), (2, 4), (3, 4), (4, 4)	3
(1e)	1:	$\{1.2, -1.5\}^T$	6	4	3	(1, 0), (0, 1)	4	all	4
	2:	$\{-0.6, 0.6\}^T$	5	3	3	(1, 0), (0, 1)	3	all	3
(1f)	1:	$\{2, 2, \ldots, 2\}^T$	6	5	4	(0, 1)	4	(3, 1), (4, 1), (2, 2), (3, 2), (4, 2), (1, 3), (2, 3), (3, 3), (4, 3), (0, 4), (1, 4), (2, 4), (3, 4), (4, 4)	3
	2:	$\{-4, -4, \ldots, -4\}^T$	7	6	5	(1, 0), (0, 1)	5	(4, 2), (3, 3), (4, 3), (2, 4), (3, 4), (4, 4)	3
(1g)	1:	$\{-1, -1, -1, -1\}^T$	6	5	4	(0, 1)	4	(3, 1), (4, 1), (1, 2), (2, 2), (3, 2), (4, 2), (1, 3), (2, 3), (3, 3), (4, 3), (0, 4), (1, 4), (2, 4), (3, 4), (4, 4)	3
	2:	$\{2, 2, 2, 0\}^T$	7	5	5	(1, 0), (0, 1)	5	(4, 2), (3, 3), (4, 3), (2, 4), (3, 4), (4, 4)	3

Members of the PM family were generated for all $(p, q) \in \{0, 1, 2, 3, 4\} \times \{0, 1, 2, 3, 4\}$.

In Table 5 and Figure 2, we present a comparison of results obtained using the proposed family of methods, PM (with orders of convergence being three and four), alongside methods from [33]: CN (order 2), Tr (order 3), and NAd (order 4). For each system, the initial estimate is specified, and the number of iterations (*iter*) required to satisfy the exit criterion is reported for each method. Results for CN, Tr, and NAd are quoted directly from [33]. Results for PM include the corresponding (p, q) pairs for which the reported number of iterations was obtained. For PM(4), "all" refers to all (p, q) pairs except (0, 1), (1, 0), and (0, 0), as these values yield third- and second-order convergence, respectively.

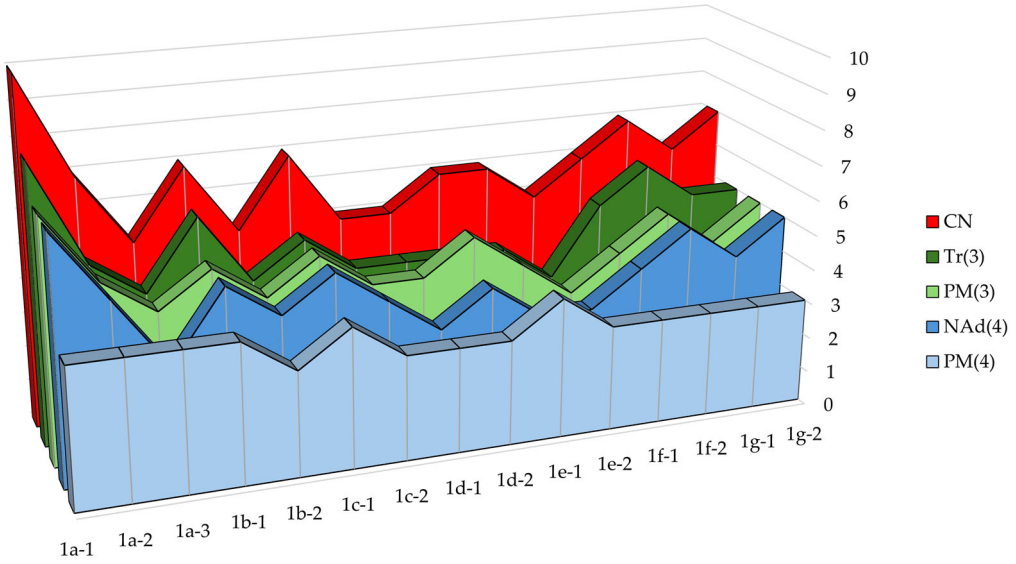


Figure 2. Comparison of iteration counts across methods and examples.

Overall, the PM methods demonstrate strong stability and require fewer iterations compared to the methods from [33], while also exhibiting higher orders of convergence. When considering only the number of iterations to reach the exit criterion, the third-order method Tr outperforms the third-order PM method in one instance (Example 1d), whereas in all other cases, PM performs equally or better. Compared to NAd, the fourth-order PM methods perform better in 10 cases, require one additional iteration in two cases, and yield identical results in the remaining cases.

Figure 2 shows the iteration counts across methods and examples, derived from the data presented in Table 5. Fourth-order methods are marked in blue, third-order methods in green, and Newton's method in red.

Example 2. *We consider nonlinear systems of equations from [35]. Example (2b) also appears in [39].*

(2a)
$$f(x) = \left\{ {x_1}^2 - x_2 - 19,\ \frac{{x_2}^3}{6} - {x_1}^2 + x_2 - 17 \right\},\ \alpha = \{5, 6\},\ x^{(0)} = \{5.1, 6.1\}$$

(2b)
$$f(x) = \left\{ \cos x_2 - \sin x_1,\ {x_3}^{x_1} - \frac{1}{x_2},\ \exp x_1 - {x_3}^2 \right\},$$
$$\alpha = \{0.9095694945\ldots, 0.6612268322\ldots, 1.5758341439\ldots\},\ x^{(0)} = \{1, 0.5, 1.5\}$$

(2c)
$$f(x) = \{x_2x_3 + x_4(x_2 + x_3), x_1x_3 + x_4(x_1 + x_3),\ x_1x_2 + x_4(x_1 + x_2),\ x_1x_2 + x_1x_3 + x_2x_3 - 1\},$$
$$\alpha = \{0.5773502691\ldots, 0.5773502691\ldots, 0.5773502691\ldots, -0.2886751345\ldots\},\ x^{(0)} = \{0.5, 0.5, 0.5, -0.2\}$$

The following stopping criterion was used:

$$Err = \left\| f\left(x^{(k)}\right) \right\|_2 + \left\| x^{(k)} - x^{(k-1)} \right\|_2 < 10^{-100}$$

Table 6 refers to methods and examples from [35], which are the sources of the reported results for methods NM, AM(3), AM(4), and NR, compared to the members of the proposed family PM(3) and PM(4).

NM refers to Newton's method. AM(3) is the third-order method free from second derivatives, defined by (2.7) in [35], while AM(4) is the improved fourth-order Arithmetic Mean Newton method given by (3.5) in [35].

NR denotes the iterative scheme defined by (1.1) in [35]. PM(3) and PM(4) are third- and fourth-order members of the proposed family, with specific values of (p, q).

Observing only the number of iterations required to satisfy the exit criterion in Table 6, the third-order method, AM, outperforms the third-order PM method in Example (2a). Similarly, in Example (2b), the fourth-order method NR performs better than the fourth-order PM method. In all other cases, PM requires fewer or equal iterations compared to AM and NR while also achieving a smaller error (*Err*).

Table 6. Comparison of iteration counts for PM(3), PM(4), and reference methods in Example 2.

Method		**Example 2a**			**Example 2b**			**Example 2c**		
	Ord	***Iter***	**(*p*, *q*)**	**Err**	***Iter***	**(*p*, *q*)**	**Err**	***Iter***	**(*p*, *q*)**	**Err**
NM	2	7		8.8 (−113)	9		1.7 (−107)	8		1.7 (−144)
AM(3)	3	5		1.9 (−140)	7		5.9 (−255)	6		6.3 (−276)
NR	4	4		8.8 (−113)	5		1.7 (−107)	5		8.9 (−291)
AM(4)	4	4		4.9 (−102)	6		2.7 (−229)	5		1.0 (−237)
PM(4)	4	4	(3, 0)	6.5 (−131)	6	(2, 1)	8.1 (−392)	4	(3, 4)	7.4 (−832)
PM(3)	3	7	(0, 1)	4.4 (−285)	7	(0, 1)	4.4 (−285)	6	(0, 1)	2.1 (−276)

Example 3. *We consider Example 1 from [17], which also appears in [3,11]:*

(3a) $$f(x) = \left\{{x_1}^2 + {x_2}^2 + {x_3}^2 - 1,\ 2{x_1}^2 + {x_2}^2 - 4x_3,\ 3{x_1}^2 - 4{x_2}^2 + {x_3}^2\right\}.$$

Performance of fourth-order methods from PM is compared to the method (2) from [17], referred to as KC, with KC1, KC2, and KC3 denoting the cases with parameter values $a = 1/2$, $a = 5/8$, and $a = 8/10$, respectively. A comparison of the results in Tables 7 and 8 highlights a favorable performance of the PM(4) methods.

Table 7. Numerical results for KC methods from [17], for example, (3a).

Method	***K***	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
		x_0**=**{**1,1,1**}		$x^{(0)}$**=**{**1/2,1/2,1/2**}	
KC1	3	1.2 (−43)	2.3 (−44)	2.2 (−47)	4.3 (−48)
	4	5.4 (−175)	1.0 (−175)	6.3 (−190)	1.2 (−190)
	5	2.2 (−700)	4.3 (−701)	4.2 (−760)	7.9 (−761)
KC2	3	9.2 (−41)	1.8 (−41)	4.1 (−69)	7.8 (−70)
	4	2.3 (−164)	4.4 (−165)	8.8 (−278)	1.7 (−278)
	5	8.9 (−659)	1.7 (−659)	1.9 (−1112)	3.6 (−1113)
KC3	3	6.5 (−34)	1.3 (−34)	1.2 (−49)	2.2 (−50)
	4	3.2 (−136)	6.1 (−137)	3.2 (−199)	6.2 (−200)
	5	1.8 (−545)	3.5 (−546)	1.9 (−797)	3.6 (−798)

Table 8. Numerical results of the PM(4) family, for example, (3a).

(*p*, *q*)	***K***	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
		x_0**=(1,1,1)**		$x^{(0)}$**=(1/2,1/2,1/2)**	
(2, 1)	3	1.3 (−53)	2.5 (−54)	2.8 (−105)	5.5 (−106)
	4	1.9 (−268)	3.6 (−269)	3.1 (−526)	6.0 (−527)
	5	1.2 (−1342)	2.2 (−1343)	1.8 (−2631)	3.5 (−2632)
(1, 2)	3	3.4 (−55)	6.5 (−56)	3.3 (−99)	6.3 (−100)
	4	1.6 (−276)	3.1 (−277)	1.3 (−496)	2.5 (−497)
	5	3.7 (−1383)	7.2 (−1384)	1.4 (−2483)	2.7 (−2484)
(3, 0)	3	2.5 (−42)	4.7 (−43)	5.5 (−48)	1.1 (−48)
	4	4.5 (−211)	8.6 (−212)	2.6 (−239)	4.9 (−240)
	5	9.1 (−1055)	1.7 (−1055)	5.3 (−1196)	1.0 (−1196)

Table 8. *Cont.*

(p, q)	K	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
		$x_0=(1,1,1)$		$x^{(0)}=(1/2,1/2,1/2)$	
(0, 3)	3	3.2 (−48)	6.2 (−49)	2.5 (−67)	4.7 (−68)
	4	6.4 (−241)	1.2 (−241)	1.6 (−336)	3.1 (−337)
	5	1.8 (−1204)	3.5 (−1205)	2.0 (−1682)	3.7 (−1683)
(3, 1)	3	1.9 (−73)	3.6 (−74)	7.8 (−119)	1.5 (−119)
	4	1.3 (−440)	2.5 (−441)	3.1 (−713)	5.9 (−714)
	5	6.9 (−2644)	1.3 (−2644)	1.2 (−4279)	2.3 (−4280)
(1, 3)	3	1.3 (−76)	2.6 (−77)	7.3 (−124)	1.4 (−124)
	4	4.4 (−460)	8.4 (−461)	1.1 (−743)	2.2 (−744)
	5	5.2 (−2761)	9.9 (−2762)	1.5 (−4462)	2.9 (−4463)
(1, 1)	3	2.0 (−32)	5.6 (−33)	4.3 (−46)	8.1 (−47)
	4	1.9 (−129)	3.7 (−130)	8.5 (−185)	1.6 (−185)
	5	3.7 (−518)	7.1 (−519)	1.3 (−739)	2.5 (−740)
(3, 2)	3	3.3 (−100)	6.3 (−101)	2.1 (−171)	4.0 (−172)
	4	5.6 (−702)	1.1 (−702)	1.6 (−1200)	3.1 (−1201)
	5	1.6 (−4914)	3.1 (−4915)	2.7 (−8404)	5.1 (−8405)
(2, 3)	3	6.0 (−102)	1.1 (−102)	1.2 (−172)	2.3 (−173)
	4	1.8 (−714)	3.5 (−715)	2.4 (−1209)	4.5 (−1210)
	5	4.9 (−5002)	9.3 (−5003)	2.7 (−8466)	5.2 (−8467)

Example 4. *We consider Example 5 from [17], also known as the Bratu problem, which also appears in [18]:*

$$x'' + Ce^{x} = 0,\ x(0) = x(1) = 0.$$

Taking $C = 3$ and $h = 1/41$, with an equidistant mesh $t_i = ih,\ i = 0, \ldots, 40$, and using the standard second-order difference scheme

$$x_k'' \approx \frac{x_{k-1} - 2x_k + x_{k+1}}{h^2}$$

we obtain the nonlinear system

(4a)
$$f(x) = \begin{cases} 2x_1 + x_2 + h^2 C \exp(x_1) \\ x_{i-1} - 2x_i + x_{i+1} + h^2 C \exp(x_i), \quad i = 2, \ldots, 39 \\ x_{39} - 2x_{40} + h^2 C \exp(x_{40}) \end{cases}.$$

Results are compared between the KC1, KC2, and KC3 methods from [17] and the PM(4) methods in the same manner as in Example 3, and are presented in Tables 9 and 10.

Table 9. Numerical results for KC methods from [17], for example, (4a).

Method	K	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
		$x^{(0)}=\{\sin(k\pi h)\}$, $k=1,\ldots,40$		$x^{(0)}=\{1/5,1/5,\ldots,1/5\}$	
KC1	3	3.2 (−42)	1.2 (−39)	5.9 (−62)	2.1 (−59)
	4	2.3 (−162)	8.4 (−160)	2.7 (−241)	9.6 (−239)
	5	6.3 (−643)	2.3 (−640)	1.1 (−958)	4.0 (−956)
KC2	3	1.0 (−57)	3.4 (−55)	2.1 (−55)	7.0 (−53)
	4	2.8 (−224)	9.3 (−222)	4.9 (−215)	1.6 (−212)
	5	1.5 (−890)	5.1 (−888)	1.4 (−853)	4.7 (−851)
KC3	3	5.3 (−41)	1.8 (−38)	1.2 (−50)	4.0 (−48)
	4	5.7 (−157)	1.9 (−154)	1.3 (−195)	4.5 (−193)
	5	7.1 (−621)	2.4 (−618)	2.0 (−775)	6.9 (−773)

Table 10. Numerical results of the PM(4) family, for example, (4a).

(p, q)	K	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
		$x^{(0)}=\{\sin(k\pi h)\}$, $k=1,\ldots,40$		$x^{(0)}=\{1/5,1/5,\ldots,1/5\}$	
(2, 1)	3	8.5 (−54)	2.9 (−51)	2.3 (−54)	7.7 (−52)
	4	1.7 (−208)	5.7 (−206)	9.0 (−211)	3.0 (−208)
	5	2.6 (−827)	8.8 (−825)	2.1 (−836)	7.1 (−834)
(1, 2)	3	3.3 (−58)	1.1 (−55)	1.4 (−54)	4.7 (−52)
	4	3.7 (−226)	1.3 (−223)	1.2 (−211)	4.0 (−209)
	5	6.2 (−898)	2.1 (−895)	6.8 (−840)	2.3 (−837)
(3, 0)	3	2.4 (−19)	8.0 (−17)	2.5 (−51)	8.4 (−49)
	4	1.0 (−70)	3.4 (−68)	1.7 (−198)	4.3 (−196)
	5	3.6 (−276)	1.2 (−273)	8.3 (−788)	2.8 (−785)
(0, 3)	3	6.5 (−34)	2.2 (−31)	3.1 (−53)	1.1 (−50)
	4	5.9 (−129)	2.0 (−126)	3.1 (−206)	1.1 (−203)
	5	3.9 (−509)	1.3 (−506)	3.1 (−818)	1.0 (−815)
(3, 1)	3	8.4 (−44)	2.8 (−41)	5.0 (−55)	1.7 (−52)
	4	1.6 (−168)	5.3 (−166)	1.9 (−213)	6.5 (−211)
	5	2.0 (−667)	6.7 (−665)	4.5 (−847)	1.5 (−844)
(1, 3)	3	1.2 (−46)	3.9 (−44)	4.2 (−55)	1.4 (−52)
	4	5.8 (−180)	1.9 (−177)	1.0 (−213)	3.3 (−211)
	5	3.5 (−713)	1.2 (−710)	3.1 (−848)	1.1 (−845)
(1, 1)	3	1.2 (−37)	4.2 (−35)	2.1 (−49)	7.4 (−47)
	4	1.9 (−143)	6.6 (−141)	1.8 (−190)	6.3 (−188)
	5	1.2 (−566)	4.0 (−564)	1.0 (−754)	3.5 (−752)
(3, 2)	3	1.5 (−49)	4.9 (−47)	3.3 (−55)	1.1 (−52)
	4	1.5 (−191)	5.1 (−189)	3.8 (−214)	1.3 (−211)
	5	1.7 (−759)	5.7 (−757)	6.7 (−850)	2.2 (−847)
(2, 3)	3	2.6 (−49)	8.6 (−47)	3.3 (−55)	1.1 (−52)
	4	1.4 (−190)	4.7 (−188)	3.7 (−214)	1.2 (−211)
	5	1.2 (−755)	4.1 (−753)	6.0 (−850)	2.0 (−847)

Example 5. *The Broyden Banded Function is often used for numerical verification of iterative methods for nonlinear systems of equations. The proposed family of methods is tested on Example 5 from [18]:*

$$F_i(x) = \left(2+5{x_i}^2\right)x_i + 1 - \sum_{j\in S(i)} x_j\left(1+2x_j\right), i = 1,2,\ldots,50$$
$$S(i) = \{j : j \neq i, \max(1, i-p) \leq j \leq \min(n, i+p)\}. \quad (5a)$$

The dimension of the system and the stopping criterion were the same as in [18]: $\left\|x^{(k)} - x^{(k-1)}\right\| < 10^{-500}$ and $\left\|F(x^{(k)})\right\| < 10^{-500}$.

The performance of several best-performing members of our family is shown in Table 11. The results are presented after three iterations and are comparable to the methods compared in Table 5 in [18]. Our family exhibits slightly better performance than the best methods in Table 5 in [18] for $(p, q) = (4, 2)$ and $(p, q) = (2, 2)$.

Table 11. Numerical results of the PM(4) family, for example, (5a).

(p, q)	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$	(p, q)	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k+1)}-x^{(k)}\right\|$
(4, 2)	1.8 (−50)	1.9 (−51)	(2, 3)	1.5 (−40)	1.2 (−41)
(2, 2)	3.1 (−49)	3.1 (−50)	(3, 2)	1.8 (−38)	1.4 (−39)
(2, 4)	5.9 (−45)	6.3 (−46)	(1, 3)	2.6 (−36)	2.0 (−37)
(3, 3)	5.3 (−45)	5.7 (−45)	(1, 2)	5.9 (−33)	4.4 (−34)
(4, 4)	3.4 (−43)	3.5 (−44)	(1, 4)	1.1 (−30)	8.0 (−32)
(3, 4)	1.0 (−42)	1.0 (−43)	(2, 1)	8.8 (−27)	6.7 (−28)
(4, 3)	1.8 (−42)	1.7 (−43)	(3, 1)	9.1 (−26)	7.0 (−27)

Example 6. *The proposed family of methods is tested on Examples 1 to 4 from [20], using identical initial estimates and stopping criterion,* $\left\|x^{(k+1)}-x^{(k)}\right\| < 10^{-150}$ *, or* $\left\|F(x^{(k)})\right\| < 10^{-150}$. *Example (6d) is also studied in [10,12,20,31,36].*

(6a)
$$\begin{cases} {x_1}^2 - x_2 - 19 = 0, \\ {x_2}^3/6 - {x_1}^2 + x_2 - 17 = 0. \end{cases}$$

(6b)
$$\begin{cases} \cos(x_2) - \sin(x_1) = 0, \\ x_1 - 1/x_2 = 0. \end{cases}$$

(6c)
$$x_i - 1.5\sin\left(\sum_{j=1, j\neq i}^{20} x_j\right) = 0, i = 1, \ldots, 20$$

(6d)
$$\begin{aligned} x_i x_{i+1} - 1 = 0, \quad i = 1, \ldots, 98 \\ x_1 x_{99} - 1 = 0. \end{aligned}$$

We present the range of parameters (p, q) that yielded the best results for each example. Notably, our family includes Jarratt's method as a special case for $(p, q) = (1, 1)$, which was also favorably compared to other methods in [20]. As shown in Tables 12–16, a specific subset of the proposed family exhibits good performance across the tested examples, both in absolute terms and when compared with Jarratt's method.

Table 12. Performance of the PM(4) methods, for example, (6a), $x^{(0)} = \{1, 2\}$.

(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)}-x^{(k-1)}\right\|$	(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)}-x^{(k-1)}\right\|$
(4, 4)	5	1.12 (−575)	2.26(−142)	(1, 1)	6	2.03 (−344)	2.48 (−86)
(3, 4)	6	2.66 (−504)	3.14(−126)	(3, 3)	5	1.41 (−272)	2.68 (−68)
(2, 3)	6	8.15 (−393)	2.34(−98)	(1, 2)	6	2.21 (−226)	9.49 (−57)
(3, 2)	7	1.86 (−386)	9.09(−97)	(2, 2)	5	2.16 (−174)	9.43 (−44)

Table 13. Comparison results of the PM(4) methods, for example, (6b), $x^{(0)} = \{1, 4\}$.

(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)}-x^{(k-1)}\right\|$	(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)}-x^{(k-1)}\right\|$
(0, 3)	5	3.22 (−482)	1.18 (−120)	(1, 4)	5	4.08 (−450)	1.26 (−112)
(2, 1)	5	2.38 (−455)	6.17 (−114)	(3, 2)	5	7.04 (−450)	1.44 (−112)
(1, 2)	5	7.27 (−453)	2.58 (−113)	(2, 3)	5	7.78 (−450)	1.48 (−112)
(4, 1)	5	8.54 (−451)	8.49 (−113)	(1, 1)	5	4.60 (−429)	2.26 (−107)

Table 14. Comparison results of the PM(4) methods, for example, (6c), $x^{(0)} = \{1.25, \ldots, 1.25\}$.

(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
(1, 2)	6	3.87 (−519)	7.18 (−131)	(4, 3)	6	1.25 (−346)	9.61 (−88)
(0, 2)	6	1.39 (−469)	1.25 (−118)	(1, 4)	6	3.65 (−329)	2.13 (−83)
(3, 3)	6	8.83 (−368)	4.96 (−93)	(4, 4)	6	1.11 (−326)	9.33 (−83)
(2, 3)	6	2.02 (−349)	1.93 (−88)	(1, 1)	6	3.01 (−204)	1.96 (−52)

Table 15. Comparison results of the PM(4) methods, for example, (6d), $x^{(0)} = \{0.1, \ldots, 0.1\}$.

(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
(4, 4)	4	1.25 (−870)	1.42 (−86)	(3, 4)	4	6.42 (−571)	5.84 (−63)
(2, 2)	5	8.32 (−677)	2.25 (−112)	(1, 1)	6	4.29 (−356)	1.14 (−88)
(4, 1)	10	9.61 (−604)	6.51 (−86)	(3, 3)	4	4.29 (−356)	4.76 (−44)
(2, 1)	7	2.62 (−572)	4.24 (−114)	(4, 3)	4	1.51 (−354)	6.26 (−39)

Table 16. Comparison results of the PM(4) methods, for example (6d), $x^{(0)} = \{-20, \ldots, -20\}$.

(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	(p, q)	Iter	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
(2, 3)	5	1.18 (−729)	8.66 (−104)	(1, 4)	5	4.17 (−603)	8.96 (−86)
(3, 2)	5	1.11 (−693)	1.15 (−98)	(2, 1)	6	4.05 (−602)	4.62 (−120)
(1, 2)	6	2.81 (−678)	2.94 (−135)	(0, 2)	5	1.06 (−528)	6.77 (−132)
(4, 0)	6	2.98 (−622)	1.49 (−103)	(1, 1)	6	3.67 (−177)	6.17 (−44)

Example 7. *The performance of the proposed family of methods was evaluated using three problems (Experiments 1 to 3) from [19]. Example (7c) also appears in [40].*

$$\text{(7a)} \quad f_i(x) = \cos(x_i) - 1, \quad i = 1, 2, \ldots, n$$

$$\text{(7b)} \quad f_i(x) = \left(1 - {x_i}^2\right) + x_i(1 + x_i x_{n-2} x_{n-1} x_n) - 2, \, i = 1, 2, \ldots, n$$

$$\text{(7c)} \quad f_i(x) = {x_i}^2 x_{i+1} - 1, \quad i = 1, 2, \ldots, n-1$$
$$f_n(x) = {x_n}^2 x_1 - 1.$$

In [19], the stopping criteria were $\left\|F(x^{(k)})\right\| < 10^{-3}$ for Experiment 1, and $\left\|F(x^{(k)})\right\| < 10^{-8}$ for Experiments 2 and 3. These criteria were satisfied after four iterations in most cases, in one case in five iterations, and after three iterations in the remaining ones, using fourth-order methods. Tables 17–19 present the results obtained with methods from the PM(4) family after four iterations, which enables comparison with the computations reported in [19]. For each example, the range of parameters (p, q) that yielded the best results is presented, including the case $(p, q) = (1, 1)$, which corresponds to Jarratt's method.

Table 17. Comparison results of PM(4) methods, for example, (7a), $x^{(0)} = \{0.78, \ldots, 0.78\}$.

(p, q)	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
	$n = 50$		$n = 100$	
(4, 4)	6.28 (−8)	8.48 (−3)	8.88 (−8)	1.20 (−2)
(3, 4)	1.19 (−7)	1.04 (−2)	1.69 (−7)	1.47 (−2)
(4, 3)	1.29 (−7)	1.07 (−2)	1.82 (−7)	1.51 (−2)
(3, 3)	2.49 (−7)	1.31 (−2)	3.53 (−7)	1.86 (−2)
(2, 4)	2.80 (−7)	1.36 (−2)	3.95 (−7)	1.93 (−2)
(4, 2)	3.40 (−7)	1.45 (−2)	4.81 (−7)	2.05 (−2)
(2, 3)	5.90 (−7)	1.73 (−2)	8.34 (−7)	2.45 (−2)
(1, 2)	5.95 (−6)	3.67 (−2)	8.42 (−6)	5.19 (−2)
(1, 1)	3.08 (−5)	6.27 (−2)	4.36 (−5)	8.86 (−2)

Table 18. Comparison results of PM(4) methods, for example, (7b), $x^{(0)} = \{2, \ldots, 2\}$.

(p, q)	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
	$n = 50$		$n = 100$	
(0, 4)	1.11 (−43)	3.66 (−11)	1.58 (−43)	5.18 (−11)
(3, 1)	5.10 (−39)	5.35 (−10)	7.21 (−39)	7.57 (−10)
(0, 3)	1.50 (−29)	1.25 (−7)	2.12 (−29)	1.76 (−7)
(1, 1)	1.97 (−24)	2.33 (−6)	2.79 (−24)	3.30 (−6)
(4, 0)	1.04 (−23)	3.60 (−6)	1.47 (−23)	5.08 (−6)
(3, 0)	2.66 (−17)	1.44 (−4)	3.76 (−17)	2.04 (−4)
(0, 2)	7.22 (−16)	2.46 (−4)	1.02 (−15)	3.49 (−4)
(1, 2)	1.31 (−15)	3.81 (−4)	1.85 (−15)	5.39 (−4)
(2, 1)	9.86 (−12)	3.55 (−3)	1.39 (−11)	5.02 (−3)

Table 19. Comparison results of the PM(4) methods, for example, (7c), $x^{(0)} = \{3, \ldots, 3\}$.

(p, q)	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$	$\left\|F(x^{(k)})\right\|$	$\left\|x^{(k)} - x^{(k-1)}\right\|$
	$n = 50$		$n = 100$	
(3, 1)	8.50 (−109)	4.16 (−27)	1.20 (−108)	5.89 (−27)
(1, 3)	1.95 (−82)	1.62 (−20)	2.76 (−82)	2.29 (−20)
(1, 2)	1.34 (−66)	1.48 (−16)	1.90 (−66)	2.09 (−16)
(2, 2)	8.81 (−63)	1.33 (−15)	1.25 (−62)	1.88 (−15)
(2, 1)	1.29 (−62)	1.46 (−15)	1.83 (−62)	2.07 (−15)
(0, 4)	1.88 (−58)	1.61 (−14)	2.66 (−58)	2.27 (−14)
(4, 1)	2.37 (−57)	3.03 (−14)	3.35 (−57)	4.28 (−14)
(0, 3)	1.17 (−43)	8.02 (−11)	1.65 (−43)	1.13 (−10)
(1, 1)	2.86 (−35)	8.43 (−9)	4.05 (−35)	1.19 (−8)

Performance in the case of large nonlinear systems is an important aspect to investigate. We tested several examples on a Windows 11 PC equipped with an Intel Core i5-1135G7 CPU (2.4 GHz) and 8 GB RAM. Example (7c) was analyzed by measuring the CPU time required for the fifth iteration, with $p = q = 1$, across varying system sizes ($n = 50, 100, \ldots, 500$) and different numerical precisions (100, 500, 1000, and 2000 digits), as shown in Figure 3. A similar dependence can be observed in all three cases.

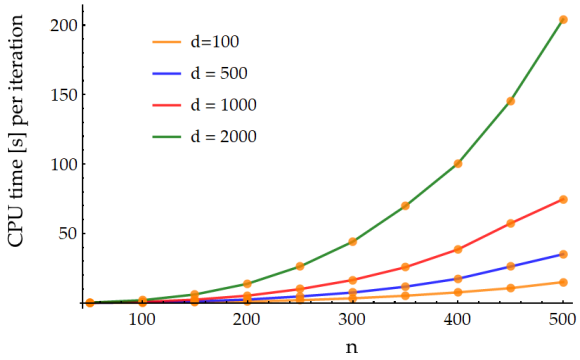


Figure 3. CPU time as a function of system size and numerical precision d.

Furthermore, we provide a comparison of the quantity $-\log\left\|F\left(x^{(k)}\right)\right\|$ among different fourth-order members of the PM family $(p+q>1)$, computed with a 2000-digit precision, and $n=100$. The results for Examples (7b) and (7c), each after five iterations, with the initial approximations $x^{(0)}=\{0.9,\ldots,0.9\}$ and $x^{(0)}=\{1.5,\ldots,1.5\}$, respectively, are shown in Figures 4 and 5. These results highlight the influence of the parameters p and q, i.e., the Padé approximations employed. In Figure 4, we observe that for larger values of the parameters p and q, the norm $\|F(x_k)\|$ does not change significantly, whereas the computational time increases.

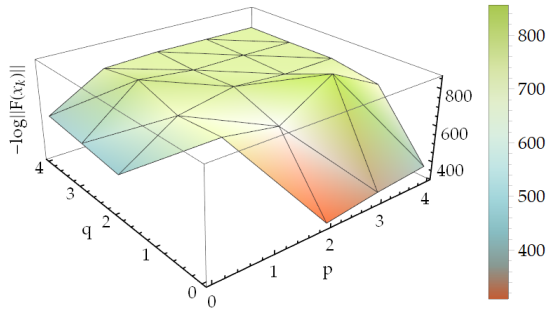


Figure 4. Comparison of $-\log\left\|F\left(x^{(5)}\right)\right\|$ for PM(4) methods applied to Example (7b).

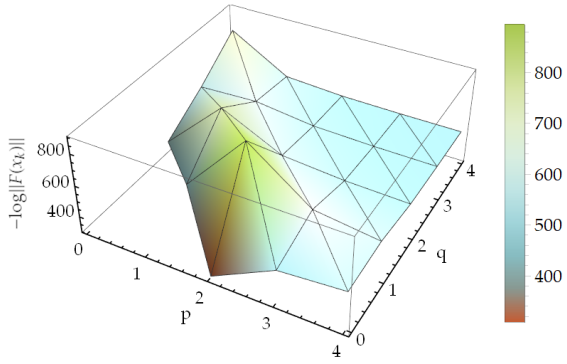


Figure 5. Comparison of $-\log\left\|F\left(x^{(5)}\right)\right\|$ for PM(4) methods applied to Example (7c).

In both cases, for each choice of the parameters p and q, all methods converged. A comparison of the results in Figures 4 and 5 indicates that, in general, no single member of the proposed family can be regarded as "best", since convergence depends on the problem, the choice of (p,q), and other parameters. It should be noted that for $(p,q)=(1,1)$, the PM family member corresponds to a generalization of the Ostrowski method [43] for systems of equations, which exhibits numerical results comparable to those of other members of the PM family. A particular case with $q=0$ is noteworthy, since the denominator in (4) reduces

to 1 and the FB operator is omitted, lowering the number of linear solves per iteration from three to two, and reducing CPU time.

Figures 6 and 7 present the ratios of CPU times for three iterations of Example (7c), computed with 1000-digit precision, using the PM(4) methods. Figure 6 shows the ratio of CPU time for $n = 100$ relative to $n = 50$, while Figure 7 shows the ratio for $n = 200$ relative to $n = 100$. Although the value of $-\log\left\|F\left(x^{(k)}\right)\right\|$ changes only slightly as the system size increases, the CPU time grows noticeably.

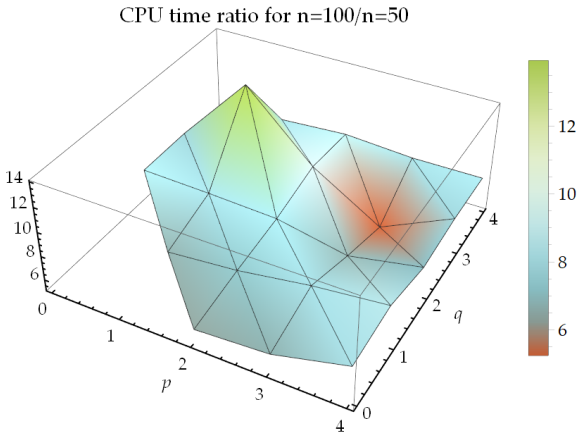


Figure 6. Ratio of CPU times for $n = 100$ compared to $n = 50$ in Example (7c).

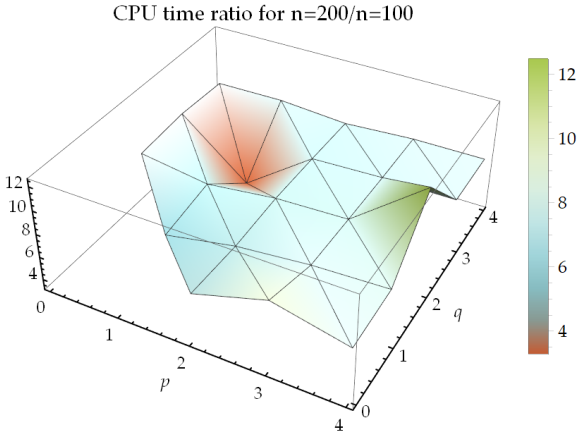


Figure 7. Ratio of CPU times for $n = 200$ compared to $n = 100$ in Example (7c).

6. Conclusions

We have presented a family of iterative methods for the numerical solution of systems of nonlinear equations. The methods for nonlinear systems are adapted from the single-variable case, where the original PM framework was first developed and analyzed. By extending the structure and convergence properties of these one-dimensional methods to higher dimensions, we preserve their theoretical foundations while enabling practical application to nonlinear systems.

These methods exhibit convergence orders ranging from second to fourth, and the numerical results align well with the theoretical analysis. Based on both theoretical insights and computational examples, the proposed methods are efficient and practical for solving nonlinear systems.

Notably, cases (1f), (4a), (5a), and (6d) demonstrate that the efficiency of the methods is preserved even for large-scale systems. Several known methods or special cases are included within our family, and many other established methods are closely related, as

illustrated throughout the paper. It is likely that additional methods may also be interpreted as members or near variants of this family.

In terms of computation time, methods with higher indices tend to require more time when applied to simpler functions such as polynomials. However, for more complex functions, most methods in the family exhibit comparable runtimes. These contrasting behaviors are exemplified in the results for solving nonlinear equations in one variable, as shown in [4].

Compared to reference methods from the literature, the PM family demonstrates favorable numerical performance. In the presented examples, PM methods require fewer or equal iterations to satisfy the exit criterion, while also achieving smaller error values. This balance of convergence order, stability, and computational efficiency highlights the practical value of the proposed family.

Author Contributions: Conceptualization, Đ.H. and D.H.; methodology, Đ.H. and D.H.; software, Đ.H. and V.T.; validation, D.H. and V.T.; formal analysis, Đ.H. and D.H.; investigation, Đ.H. and D.H.; data curation, V.T. and P.O.; writing—original draft preparation, Đ.H. and D.H.; writing—review and editing, Đ.H., D.H., V.T. and P.O.; visualization, Đ.H., D.H., V.T. and P.O.; supervision, Đ.H. and P.O.; project administration, D.H. and V.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science, Technological Development and Innovation of the Republic of Serbia under projects 451-03-137/2025-03/200125, 451-03-136/2025-03/200125, and 451-03-137/2025-03/200156.

Data Availability Statement: The original contributions presented in this study are included in the article. For further inquiries, please contact the corresponding author.

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

References

1. Herceg, Đ.; Herceg, D. A family of methods for solving nonlinear equations. *Appl. Math. Comput.* **2015**, *259*, 882–895. [CrossRef]
2. Jarratt, P. Some fourth-order multipoint iterative methods for solving equations. *Math. Comput.* **1966**, *20*, 434–437. [CrossRef]
3. Yaseen, S.; Zafar, F. A new sixth-order Jarratt-type iterative method for systems of nonlinear equations. *Arab. J. Math.* **2022**, *11*, 585–599. [CrossRef]
4. Babajee, D.K.R.; Madhu, K.; Jayaraman, J. A family of higher order multipoint iterative methods based on power mean for solving nonlinear equations. *Afr. Mat.* **2016**, *27*, 865–876. [CrossRef]
5. Li, S.; Liu, X.; Zhang, X. A Few Iterative Methods by Using [1, n]-Order Padé Approximation of Function and the Improvements. *Mathematics* **2019**, *7*, 55. [CrossRef]
6. Özban, A.Y. Some new variants of Newton's method. *Appl. Math. Lett.* **2004**, *17*, 677–682. [CrossRef]
7. Kou, J. Fourth-order variants of Cauchy's method for solving non-linear equations. *Appl. Math. Comput.* **2007**, *192*, 113–119. [CrossRef]
8. Kou, J.; Li, Y.; Wang, X. Fourth-order iterative methods free from second derivative. *Appl. Math. Comput.* **2007**, *184*, 880–885. [CrossRef]
9. Frontini, M.; Sormani, E. Some variants of Newton's method with third-order convergence. *Appl. Math. Comput.* **2003**, *140*, 419–426. [CrossRef]
10. Esmaeili, H.; Ahmadi, M. An efficient three-step method to solve system of nonlinear equations. *Appl. Math. Comput.* **2015**, *266*, 1093–1101. [CrossRef]
11. Singh, H.; Sharma, J.R. Simple and efficient fifth-order solvers for systems of nonlinear problems. *Math. Model. Anal.* **2023**, *28*, 527–543. [CrossRef]
12. Wang, X.; Sun, M. A new family of fourth-order Ostrowski-type iterative methods for solving nonlinear systems. *AIMS Math.* **2024**, *9*, 10255–10266. [CrossRef]
13. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. Increasing the convergence order of an iterative method for nonlinear systems. *Appl. Math. Lett.* **2012**, *25*, 2369–2374. [CrossRef]
14. Babajee, D.K.R.; Madhu, K.; Jayaraman, J. On some improved harmonic mean Newton-like methods for solving systems of nonlinear equations. *Algorithms* **2015**, *8*, 895–909. [CrossRef]

15. Cordero, A.; Maimó, J.G.; Rodríguez-Cabral, A.; Torregrosa, J.R. Two-step fifth-order efficient Jacobian-free iterative method for solving nonlinear systems. *Mathematics* **2024**, *12*, 3341. [CrossRef]
16. Quinga, S.; Pavon, W.; Ortiz, N.; Calvopiña, H.; Yépez, G.; Quinga, M. Numerical solution of the nonlinear convection–diffusion equation using the fifth-order iterative method by Newton–Jarratt. *Mathematics* **2025**, *13*, 1164. [CrossRef]
17. Kansal, M.; Cordero, A.; Bhalla, S.; Torregrosa, J.R. New fourth- and sixth-order classes of iterative methods for solving systems of nonlinear equations and their stability analysis. *Numer. Algorithms* **2021**, *87*, 1017–1060. [CrossRef]
18. Bhalla, S.; Singh, G.; Ramos, H.; Behl, R.; Alshehri, H. A Fourth-Order Parametric Iterative Approach for Solving Systems of Nonlinear Equations. *Computation* **2025**, *13*, 241. [CrossRef]
19. Al-Obaidi, R.H.; Darvishi, M.T. Constructing a Class of Frozen Jacobian Multi-Step Iterative Solvers for Systems of Nonlinear Equations. *Mathematics* **2022**, *10*, 2952. [CrossRef]
20. Cordero, A.; Jordán, C.; Sanabria-Codesal, E.; Torregrosa, J.R. Design, Convergence and Stability of a Fourth-Order Class of Iterative Methods for Solving Nonlinear Vectorial Problems. *Fractal Fract.* **2021**, *5*, 125. [CrossRef]
21. Cordero, A.; Rojas-Hiciano, R.V.; Torregrosa, J.R.; Vassileva, M.P. Maximally efficient damped composed Newton-type methods to solve nonlinear systems of equations. *Appl. Math. Comput.* **2025**, *492*, 129231. [CrossRef]
22. Cordero, A.; Torregrosa, J.R. Variants of Newton's method for functions of several variables. *Appl. Math. Comput.* **2006**, *183*, 199–208. [CrossRef]
23. Homeier, H.H.H. A modified Newton method with cubic convergence: The multivariable case. *J. Comput. Appl. Math.* **2004**, *169*, 161–169. [CrossRef]
24. Noor, M.A.; Wasteem, M. Some iterative methods for solving a system of nonlinear equations. *Comput. Math. Appl.* **2009**, *57*, 101–106. [CrossRef]
25. Frontini, M.; Sormani, E. Third-order methods from quadrature formulae for solving systems of nonlinear equations. *Appl. Math. Comput.* **2004**, *149*, 771–782. [CrossRef]
26. Grau-Sanchez, M.; Grau, A.; Noguera, M. On the computational efficiency index and some iterative methods for solving systems of nonlinear equations. *J. Comput. Appl. Math.* **2011**, *236*, 1259–1266. [CrossRef]
27. Khirallah, M.Q.; Hafiz, M.A. Novel third-order methods for solving a system of nonlinear equations. *Bull. Math. Sci. Appl.* **2012**, *2*, 1–14. [CrossRef]
28. Darvishi, M.T.; Barati, A. Super cubic iterative methods to solve systems of nonlinear equations. *Appl. Math. Comput.* **2007**, *188*, 1678–1685. [CrossRef]
29. Xiao, X.; Yin, H. A new class of methods with higher order of convergence for solving systems of nonlinear equations. *Appl. Math. Comput.* **2015**, *264*, 300–309. [CrossRef]
30. Liu, Z.; Fang, Q. A new Newton-type method with third order for solving systems of nonlinear equations. *J. Appl. Math. Phys.* **2015**, *3*, 1256–1261. [CrossRef]
31. Cordero, A.; Torregrosa, J.R. Variants of Newton's method using fifth-order quadrature formulas. *Appl. Math. Comput.* **2007**, *190*, 686–698. [CrossRef]
32. Darvishi, M.T.; Barati, A. A third-order Newton-type method to solve systems of nonlinear equations. *Appl. Math. Comput.* **2007**, *187*, 630–635. [CrossRef]
33. Cordero, A.; Martínez, E.; Torregrosa, J.R. Iterative methods of order four and five for systems of nonlinear equations. *J. Comput. Appl. Math.* **2009**, *231*, 541–551. [CrossRef]
34. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. A modified Newton–Jarratt's composition. *Numer. Algorithms* **2010**, *55*, 87–99. [CrossRef]
35. Babajee, D.K.R.; Cordero, A.; Soleymani, F.; Torregrosa, J.R. On a novel fourth-order algorithm for solving systems of nonlinear equations. *J. Appl. Math.* **2012**, *1*, 165452. [CrossRef]
36. Sharma, J.R.; Guha, R.K.; Sharma, R. An efficient fourth-order weighted-Newton method for systems of nonlinear equations. *Numer. Algorithms* **2013**, *62*, 307–323. [CrossRef]
37. Narang, M.; Bhatia, S.; Kanwar, V. New two-parameter Chebyshev–Halley-like family of fourth and sixth-order methods for systems of nonlinear equations. *Appl. Math. Comput.* **2016**, *275*, 394–403. [CrossRef]
38. Darvishi, M.T.; Barati, A. A fourth-order method from quadrature formulae to solve systems of nonlinear equations. *Appl. Math. Comput.* **2007**, *188*, 257–261. [CrossRef]
39. Madhu, K.; Jayaraman, J. Some higher-order Newton-like methods for solving system of nonlinear equations and its applications. *Int. J. Appl. Comput. Math.* **2017**, *3*, 2213–2230. [CrossRef]
40. Lotfi, T.; Bakhtiari, P.; Cordero, A.; Mahdiani, K.; Torregrosa, J.R. Some new efficient multipoint iterative methods for solving nonlinear systems of equations. *Int. J. Comput. Math.* **2015**, *92*, 1921–1934. [CrossRef]
41. Liu, Z.; Zheng, Q.; Huang, C.-E. Third- and fifth-order Newton–Gauss methods for solving nonlinear equations with n variables. *Appl. Math. Comput.* **2016**, *290*, 250–257. [CrossRef]

42. Moscoso-Martínez, M.; Chicharro, F.I.; Cordero, A.; Torregrosa, J.R. Performance of a New Sixth-Order Class of Iterative Schemes for Solving Non-Linear Systems of Equations. *Mathematics* **2023**, *11*, 1374. [CrossRef]
43. Ostrowski, A.M. *Solution of Equations and Systems of Equations*; Academic Press Inc.: New York, NY, USA, 1966.

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.