Mathematical Modelling and Analysis
Volume 28, Issue 1, 1–22, 2023
https://doi.org/10.3846/mma.2023.16244

Simple and Efficient Fifth Order Solvers for Systems of Nonlinear Problems

Harmandeep Singh and Janak Raj Sharma

Department of Mathematics, Sant Longowal Institute of Engineering and Technology
Longowal, 148106 Punjab, India
E-mail(*corresp.*): jrshira@yahoo.co.in
E-mail: harman85pau@gmail.com

Received January 6, 2022; revised October 28, 2022; accepted November 4, 2022

Abstract. In this study, two multi-step iterative techniques of fifth order convergence are explored to solve nonlinear equations. The techniques are designed with the prime objective of keeping the computational cost as low as possible. To claim this objective, the efficiency indices are determined and compared with the efficiencies of the existing techniques of same order. The outcome of comparison analysis is remarkable from the view of high computational efficiency of new methods. Performance and stability are illustrated by executing the numerical tests on some nonlinear problems of diverse nature. The entire analysis significantly favors the new techniques compared to their existing counterparts, especially for the case of large dimensional systems.

Keywords: nonlinear systems, iterative techniques, convergence order, computational complexity.

AMS Subject Classification: 65H10; 65J10; 41A25; 49M15.

1 Introduction

The mathematical models representing the vital aspects of the physical phenomena in the field of science, engineering, etc. are inherently nonlinear in nature. A particular model, so constructed, usually describes a phenomenon by a set of equations involving relationship among a set of variables. Mathematically, a nonlinear model is generally expressed as,

$$F(x) = O, \tag{1.1}$$

Copyright © 2023 The Author(s). Published by Vilnius Gediminas Technical University
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

where, $O \in \mathbb{R}^m$ represents the zero vector, $F : \Omega \subseteq \mathbb{R}^m \to \mathbb{R}^m$ is a nonlinear mapping, commonly represented as $(f_1(x), f_2(x), \dots, f_m(x))^T$, $f_i : \mathbb{R}^m \to \mathbb{R}$ $(i = 1, \dots, m)$ are nonlinear scalar functions, and $x = (x_1, \dots, x_m)^T \in \Omega$.

To analyze the behavior of a physical phenomenon in diverse conditions, the fundamental requirement is to obtain the solution of corresponding nonlinear model. In general, obtaining the analytical or closed form solution of nonlinear models is not feasible, but in such cases, the iterative techniques can provide the solution [6, 11] in numerical form up to the desired accuracy. The working mechanism of iterative techniques is based on the theory of fixed point iterations. Under some assumptions, the solution, $x^* \in \Omega$, of the Equation (1.1) can be obtained as a fixed point of some suitable function $\psi : \mathbb{R}^m \to \mathbb{R}^m$, such that

$$x^{(n+1)} = \psi(x^{(n)}), \quad n = 0, 1, 2, \dots,$$

where $x^{(0)}$ is the initial approximation to the solution.

The Newton's technique is the most widely used iterative procedure to find solution of nonlinear models, and it approximates solution of (1.1) iteratively as,

$$x^{(n+1)} = \psi(x^{(n)}) = x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}), \quad n = 0, 1, 2, \dots, \tag{1.2}$$

where, $F'(x) \in \mathcal{L}(\mathbb{R}^m, \mathbb{R}^m)$ is a linear operator, and is generally represented as a Jacobian matrix $\left[\frac{\partial f_i}{\partial x_j}\right]_{m \times m}$. Under the hypotheses that $F(x)$ is continuous and differentiable in some neighborhood of its simple solution and that initial approximation is chosen sufficiently close to the solution, Newton's technique shows quadratic convergence. That simply means the number of significant digits double with the progress of each iteration.

Numerous iterative techniques have been developed (see, for example [3, 4, 7, 8, 10, 13, 14, 15, 16, 17], and references therein) to improve the convergence rate of the Newton's technique by introducing the additional functional or Jacobian evaluations. Evidently, Newton's technique utilizes a function evaluation (F), a Jacobian matrix (F'), and a matrix inversion (F'^{-1}) per iteration. At the cost of an additional function evaluation, the third order Potra and Pták [7] technique is the most simple improvement over the Newton's technique, which is presented below,

$$x^{(n+1)} = y^{(n)} - F'(x^{(n)})^{-1}F(y^{(n)}), \tag{1.3}$$

where $y^{(n)}$ is the Newton's iteration given by (1.2). In addition, the modified techniques developed in [8, 13] require two evaluations each of F, F', and F'^{-1}, the techniques in [3, 10] require two evaluations each of F, F', and three F'^{-1}, the techniques in [4, 15] involve evaluations of two F, four F', and three F'^{-1}, the technique in [14] requires evaluations of three F, and two each of F' and F'^{-1}, whereas the technique in [16] requires evaluations of three F, two F', and one F'^{-1}. Apart from these, some hybrid algorithms have also been developed that merge the iterative techniques together with the optimization algorithms to generate new algorithms with the improved convergence rate (see, for example [9], and references therein).

Undoubtedly, these additional evaluations significantly affect the cost of computation in terms of mathematical operations. Therefore, the development and utilization of computationally efficient techniques have gained much importance in recent times. The concept of computational efficiency index is established (see [6, 11]) to analyze and compare the efficiencies of iterative techniques. Moreover, the necessary parameters have been introduced for the rigorous investigation of this concept in [8]. Keeping into mind the challenge of optimizing the computational cost with the increasing order of convergence, here we shall present two simple yet highly efficient iterative techniques with fifth order of convergence.

We summarize here the contents of the rest of paper. Section 2 includes the development and convergence analysis of the new iterative techniques. Computational efficiency for the new techniques is determined in Section 3, and the same is analyzed and compared with the existing fifth order techniques. Numerical testing is executed in Section 4 to certify the theoretical deductions. Concluding remarks are given in Section 5.

2 Development of methods

In what follows, we shall propose a three step iterative scheme involving some undetermined parameters. The parameters' values to be chosen optimally so as to maximize the rate or order of convergence. We shall show that the proposed scheme converges with the order five. The intention here is to develop a technique which accelerates the convergence rate of Potra-Pták method (1.3) without requiring any additional inverse operators so that the computational cost may be as small as possible. In view of this, it will be judicious to consider an iterative technique of the type,

$$\begin{aligned}
y^{(n)} &= x^{(n)} - a\ F'(x^{(n)})^{-1}F(x^{(n)}),\\
z^{(n)} &= y^{(n)} - b\ F'(x^{(n)})^{-1}F(y^{(n)}),\\
x^{(n+1)} &= z^{(n)} - (c\ I + d\ F'(x^{(n)})^{-1}F'(y^{(n)}))F'(x^{(n)})^{-1}F(z^{(n)}),
\end{aligned} \tag{2.1}$$

where a, b, c, and d are the parameters to be determined in the sequel.

In order to analyze the convergence rate, we first state a preliminary result (see [5]) as a lemma which is followed by the main theorem to prove the fifth order of convergence of the aforementioned technique.

Lemma 1. *Assume that $F : \Omega \subseteq \mathbb{R}^m \to \mathbb{R}^m$ is a p-times Fréchet differentiable function in an open convex set $\Omega \in \mathbb{R}^m$, then for any $x,\ t \in \Omega$, the following result holds:*

$$F(x+t) = F(x) + F'(x)t + \frac{1}{2!}F''(x)t^2 + \ldots + \frac{1}{(p-1)!}F^{(p-1)}(x)t^{(p-1)} + R_p,$$

where $t^i = (t, \overset{i-times}{\cdots\cdots}, t)$, $F^{(i)}(x) \in \mathcal{L}(\mathbb{R}^m \times \overset{i-times}{\cdots\cdots} \times \mathbb{R}^m, \mathbb{R}^m)$ for each $i = 1, 2, \ldots,$ and

$$\|R_p\| \leq \frac{1}{p!} \sup_{0<h<1} \|F^{(p)}(x+ht)\|\ \|t\|^p.$$

As a consequence of above lemma, the Taylor expansion of $F(x)$, in the neighborhood of its zero x^*, can be written as

$$F(x^* + h) = F'(x^*) \left[h + \sum_{k=2}^{p} \frac{1}{k!} F'(x^*)^{-1} F^{(k)}(x^*) h^k \right] + O(h^{p+1}),$$

where $h \in \mathbb{R}^m$, $F'(x^*)^{-1} \in \mathcal{L}(\mathbb{R}^m, \mathbb{R}^m)$, and the k-th differentiation of $F(x)$ is a k-linear function such that $F^{(k)}(x^*)h^k = F^{(k)}(x^*)(h, \ldots, h) \in \mathbb{R}^m$. Also, $O(h^{p+1})$ represents the terms containing h^k, where $k \geq p + 1$. The operations in above equation are well-defined as it can be easily verified that each of the term in the equation belongs to $\mathbb{R}^m$. Moreover, we have

$$F'(x^* + h) = F'(x^*) \left[I + \sum_{k=2}^{p} \frac{1}{(k-1)!} F'(x^*)^{-1} F^{(k)}(x^*) h^{(k-1)} \right] + O(h^p),$$

where, $F^{(k)}(x^*)h^{k-1} \in \mathcal{L}(\mathbb{R}^m, \mathbb{R}^m)$.

Theorem 1. *Suppose that a nonlinear mapping, $F : \Omega \subseteq \mathbb{R}^m \to \mathbb{R}^m$, is differentiable sufficient number of times in some neighborhood of its simple zero, x^*, contained in an open convex set Ω. Further, assume that $F'(x)$ is nonsingular and continuous in that neighborhood, and the initial estimate, $x^{(0)}$, is sufficiently close to x^*. Then, the sequence of iterates generated by the iterative scheme* (2.1) *converges to x^* with the fifth order of convergence, provided $a = 1$, $c = 2$, $d = -1$, and either $b = 1$ or $b = -5$.*

Proof. Let $e^{(n)} = x^{(n)} - x^*$ be the local error generated at n^{th} iteration of (2.1). Using the fact that $F(x^*) = O$, then as a consequence of Lemma 1, the Taylor expansions of $F(x^{(n)})$ and $F'(x^{(n)})$, about x^*, can be established as

$$F(x^{(n)})=F'(x^*)[e^{(n)}+A_2e^{(n)^2}+A_3e^{(n)^3}+A_4e^{(n)^4}+A_5e^{(n)^5}]+O(e^{(n)^6}), \qquad (2.2)$$
$$F'(x^{(n)}) = F'(x^*)[I + 2A_2e^{(n)} + 3A_3e^{(n)^2} + 4A_4e^{(n)^3} + 5A_5e^{(n)^4}] + O(e^{(n)^5}),$$

where $e^{(n)^i} = (e^{(n)}, \ldots, e^{(n)})$, $A_i = \frac{1}{i!}F'(x^*)^{-1}F^{(i)}(x^*)$, $i = 2, 3, \ldots$, and therefore,

$$F'(x^{(n)})^{-1}=[I + B_1e^{(n)}+B_2e^{(n)^2}+B_3e^{(n)^3}+B_4e^{(n)^4}]F'(x^*)^{-1}+O(e^{(n)^5}), \quad (2.3)$$

where $B_1 = -2A_2$, $B_2 = -3A_3 + 4A_2^2$, $B_3 = -4A_4 + 6A_2A_3 + 6A_3A_2 - 8A_2^3$, $B_4 = -5A_5 + 8A_2A_4 + 9A_3^2 + 8A_4A_2 - 12A_2^2A_3 - 12A_2A_3A_2 - 12A_3A_2^2 + 16A_2^4$.

Let $e_y^{(n)} = y^{(n)} - x^*$ be the local error at the first step of (2.1). Then, using the expressions of (2.2)–(2.3) in the first step of (2.1), and after simplifying, we have that

$$e_y^{(n)} = C_1e^{(n)} + C_2e^{(n)^2} + C_3e^{(n)^3} + C_4e^{(n)^4} + C_5e^{(n)^5} + O(e^{(n)^6}), \qquad (2.4)$$

where $C_1 = 1-a$, $C_2 = aA_2$, $C_3 = 2a(A_3-A_2^2)$, $C_4 = a(3A_4-4A_2A_3-3A_3A_2+4A_2^3)$, $C_5 = a(4A_5-6A_2A_4-6A_3^2-4A_4A_2+8A_2^2A_3+6A_2A_3A_2+6A_3A_2^2-8A_2^4)$.

Using the Equation (2.4), the Taylor developments of $F(y^{(n)})$, $F'(y^{(n)})$ about x^* are given as

$$F(y^{(n)})=F'(x^*)[K_1e^{(n)}+K_2e^{(n)^2}+K_3e^{(n)^3}+K_4e^{(n)^4}+K_5e^{(n)^5}]+O(e^{(n)^6}), \tag{2.5}$$

$$F'(y^{(n)}) = F'(x^*)[I + L_1e^{(n)} + L_2e^{(n)^2} + L_3e^{(n)^3} + L_4e^{(n)^4}] + O(e^{(n)^5}), \tag{2.6}$$

where $K_1 = 1 - a$, $K_2 = (1 - a + a^2)A_2$, $K_3 = (1 - a + 3a^2 - a^3)A_3 - 2a^2A_2^2$, $K_4 = (1-a+6a^2-4a^3+a^4)A_4-4a^2A_2A_3 - 3a^2(2 - a)A_3A_2 + 5a^2A_2^3$, $K_5 = (1 - a + 10a^2 - 10a^3 + 5a^4 - a^5)A_5 - 6a^2A_2A_4 - 6a^2(2 - a)A_3^2 - 4a^2(3 - 3a + a^2)A_4A_2+10a^2A_2^2A_3+8a^2A_2A_3A_2+3a^2(5-3a)A_3A_2^2-12a^2A_2^4$, $L_1 = 2(1-a)A_2$, $L_2 = 3(1 - 2a + a^2)A_3 + 2aA_2^2$, $L_3 = 4(1 - 3a + 3a^2 - a^3)A_4 + 4aA_2A_3 + 6a(1 - a)A_3A_2 - 4aA_2^3$, $L_4 = 5(1 - 4a + 6a^2 - 4a^3 + a^4)A_5 + 6aA_2A_4 + 12a(1 - a)A_3^2 + 12a(1 - 2a + a^2)A_4A_2 - 8aA_2^2A_3 - 6aA_2A_3A_2 - 3a(4 - 5a)A_3A_2^2 + 8aA_2^4$.

Denoting $e_z^{(n)} = z^{(n)} - x^*$ as the local error at the second step of method (2.1), and using the Equations (2.3) and (2.5), the second step of (2.1) yields,

$$e_z^{(n)} = M_1e^{(n)} + M_2e^{(n)^2} + M_3e^{(n)^3} + M_4e^{(n)^4} + M_5e^{(n)^5} + O(e^{(n)^6}), \tag{2.7}$$

where $M_1 = (1-a)(1-b)$, $M_2 = (a+b-ab-a^2b)A_2$, $M_3 = (2a+2b-2ab-3a^2b + a^3b)A_3 + 2(ab - a - b + 2a^2b)A_2^2$, and $M_4 = (3a - 3ab + 3b - 6a^2b + 4a^3b - a^4b)A_4 - 2(2a - 2ab + 2b - 5a^2b + a^3b)A_2A_3 - 3(a + b - ab - 3a^2b + a^3b)A_3A_2 + (4a - 4ab + 4b - 13a^2b)A_2^3$.

Here, the expression of M_5, being lengthy, is not shown explicitly. Taylor expansion of $F(z^{(n)})$, using the Equation (2.7), is developed as

$$F(z^{(n)}) = F'(x^*)[P_1e^{(n)}+P_2e^{(n)^2}+P_3e^{(n)^3}+P_4e^{(n)^4}+P_5e^{(n)^5}]+O(e^{(n)^6}), \tag{2.8}$$

where $P_1 = (1 - a)(1 - b)$, $P_2 = (1 - b + b^2 - a + 3ab - 2ab^2 + a^2 - 3ba^2 + a^2b^2)A_2$, $P_3 = (1-b+3b^2-b^3-a+7ab-9ab^2+3ab^3+3a^2-12a^2b+9a^2b^2-3a^2b^3-a^3+4a^3b-3a^3b^2+a^3b^3)A_3-2(a^2-3a^2b+2ab-2ab^2-a^3b+a^3b^2+b^2)A_2^2$, and the explicit forms of P_4 and P_5 are being avoided here for their lengthy expressions.

Consequently, the error equation at the $(n + 1)^{th}$ iteration is obtained by substituting the expressions of (2.3), (2.6), and (2.8) in the final step of (2.1), which is given by,

$$e^{(n+1)}=x^{(n+1)}-x^*=Q_1e^{(n)}+Q_2e^{(n)^2}+Q_3e^{(n)^3}+Q_4e^{(n)^4}+Q_5e^{(n)^5}+O(e^{(n)^6}), \tag{2.9}$$

where $Q_1 = (1 - a)(1 - b)(1 - c - d)$, $Q_2 = (c + d + b(1 - c - d) - b^2(c + d) + a(1-b)(1-c-2bc+d-2bd)-a^2(c+b(1-3c-5d)+3d+b^2(c+d)))A_2$, $Q_3 = (2b+(2-2b-3b^2+b^3)(c+d)+a(1-b)(2+c(3b^2-6b-2)+d(3b^2-6b+4)) + a^2(a-3)(c+b(1-4c-7d)+4d+b^2(3-b)(c+d)))A_3+2(-b+(2b^2+b-1)(c+d)+a^2(b(2-6c-8d)-b(c+4d))+a(-1+c-3d-b^2(4c+3d)+b(1+3c+7d)))A_2^2$, and the expressions of Q_4 and Q_5 are not shown explicitly here.

Finally, the values of parameters should be selected in the sense that the scheme (2.1) attains maximum possible order of convergence. Accordingly, the coefficients Q_1, Q_2, Q_3, and Q_4 in the Equation (2.9) vanish for two sets of parametric values. For $a = b = 1$, $c = 2$, and $d = -1$, the error equation is reduced to,

$$e^{(n+1)} = 12A_2^4 {e^{(n)}}^5 + O({e^{(n)}}^6),$$

whereas, for the values $a = 1$, $b = -5$, $c = 2$, and $d = -1$, the error equation becomes,

$$e^{(n+1)} = 12(3A_3A_2^2 - A_2A_3A_2 - 3A_2^4){e^{(n)}}^5 + O({e^{(n)}}^6).$$

Hence, the fifth order of convergence is proved for the two sets of parametric values. □

The proposed fifth order iterative techniques are finally presented below.

Technique-1: For $a = b = 1$, $c = 2$ and $d = -1$, the scheme (2.1) is expressed as

$$\begin{aligned} y^{(n)} &= x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}), \quad z^{(n)} = y^{(n)} - F'(x^{(n)})^{-1}F(y^{(n)}), \\ x^{(n+1)} &= z^{(n)} - (2I - F'(x^{(n)})^{-1}F'(y^{(n)}))F'(x^{(n)})^{-1}F(z^{(n)}). \end{aligned} \tag{2.10}$$

Technique-2: For $a = 1$, $b = -5$, $c = 2$ and $d = -1$, we have the scheme (2.1) as

$$\begin{aligned} y^{(n)} &= x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}), \quad z^{(n)} = y^{(n)} + 5F'(x^{(n)})^{-1}F(y^{(n)}), \\ x^{(n+1)} &= z^{(n)} - (2I - F'(x^{(n)})^{-1}F'(y^{(n)}))F'(x^{(n)})^{-1}F(z^{(n)}). \end{aligned} \tag{2.11}$$

Clearly, both of the techniques utilize three functional evaluations, two Jacobian matrices, and one matrix inversion per iteration. The techniques (2.10) and (2.11) are denoted as $\boldsymbol{T_1}$ and $\boldsymbol{T_2}$, respectively, for the further reference.

Remark 1. It can be observed that the general parametric family of three-step iterative scheme developed by Zhanlav et al. in [16] includes (2.10) as a particular case.

3 Computational complexity

The term computational complexity refers to the analysis of an algorithm's characteristics that how much computational resources it utilizes during its course of action. Thus, for the development of an algorithm, achieving higher rate of convergence should not be the only target, but the element of efficiency must be accounted as a crucial factor. In practice, solving the systems of nonlinear equations involve large number of mathematical or numerical operations, and therefore, an efficient algorithm should be the ideal choice for the implementation of this process.

The objective here is to thoroughly investigate the developed iterative algorithms on the subject of computational efficiency. For that purpose, number of definitions are available in the literature. Defined in any way, it always shows positive relation with the convergence order (r) and negative relation with the number of mathematical operations per iteration i.e. cost of computation (C). Ostrowski [6] proposed the measure of efficiency as,

$$E_{O_1} = \log r/C, \text{ and } E_{O_2} = r^{1/C},$$

whereas Traub [11] considered it straightforwardly as, $E_T = r/C$.

In order to utilize the concept of efficiency in a rigorous way, and further, to compare the efficiencies of iterative techniques, we shall describe here the convenient approach for the estimation of computational efficiency. To locate the solution, x^*, of a system of nonlinear equations, the initial approximation, $x^{(0)}$, is selected in some neighborhood of the solution. The stopping criterion for iterations is generally prescribed as,

$$\|x^{(n)} - x^*\| \leq \delta = 10^{-d},$$

where n is the iteration index, δ is the desired precision for approximate solution $x^{(n)}$, and d is the number of significant decimal digits of that approximation. Conveniently, we assume that $\|x^{(0)} - x^*\| \approx 10^{-1}$. Then, the number of steps for an iterative procedure, which are required to reach the prescribed accuracy, can be estimated in the analytical way from the approximation, $10^{-d} \approx 10^{-r^n}$ or $n \approx \log d/\log r$, where r is the convergence order. As already discussed that the computational efficiency is reciprocally proportional to the total computational cost, nC, of the completed iterative process comprising of n iterative steps, the computational efficiency or more conventionally the efficiency index of an iterative scheme can be estimated as,

$$E = \frac{1}{nC} = \frac{1}{\log d}\frac{\log r}{C}. \tag{3.1}$$

Consider a function $F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T$, where $x = (x_1, x_2, \ldots, x_m)^T_,$ then the estimation of computational cost (see [8]) per iteration for an iterative technique can be obtained by,

$$C(m, \nu_0, \nu_1, l) = N_0(m)\nu_0 + N_1(m)\nu_1 + N(m, l), \tag{3.2}$$

where $N_0(m)$ and $N_1(m)$ represent the number of evaluations of scalar functions in the computation of F and F', respectively, and $N(m, l)$ corresponds to the number of product or quotient operations per iteration. In order to estimate $C(m, \nu_0, \nu_1, l)$ in units of products, it is essential to evaluate the ratios $\nu_0 > 0$ and $\nu_1 > 0$, which express relation between the products and evaluations, and a ratio $l > 1$, relating products and quotients.

For the suitable application of definition (3.1) along with (3.2), it is necessary to assess all the relevant elements which constitute the total numerical operational cost in the entire process. In particular, the computation of a function F involves evaluation of m scalar functions in any iteration, while m^2 scalar functions are required to compute a derivative F'. Additionally, the technique of LU decomposition is employed, to compute a inverse linear operator, followed by the resolution of two triangular linear systems to finally work out $F'^{-1}F$. Note that, LU decomposition involves $m(m-1)(2m-1)/6$ products and $m(m-1)/2$ quotients, whereas $m(m-1)$ products and m quotients are required for the resolution of two triangular systems. Moreover, m products for scalar-vector multiplication, and m^2 products for matrix-vector multiplication must be taken into the account.

The above discussion is fairly adequate to enable us to understand the concept of computational efficiency, and therefore, we now present the comparison analysis in this context. For this purpose, we include the existing fifth order techniques developed by Cordero-Hueso-Martínez-Torregrosa [3], Liu-Zheng-Huang [4], Sharma-Gupta [8], Solaiman-Hashim [10], Xiao-Yin [13], Xiao-Yin [14], Xu-Jieqing [15], and Zhanlav-Chun-Otgondorj-Ulziibayar [16]. For the ready reference, these techniques are expressed below, which are denoted by T_i, where $i = 3, 4, \dots$.

Technique by Cordero et al. (T_3):

$$\begin{aligned}&y^{(n)}=x^{(n)}-F'(x^{(n)})^{-1}F(x^{(n)}),\ z^{(n)}=x^{(n)}-2[F'(x^{(n)})+F'(y^{(n)})]^{-1}F(x^{(n)}),\\&x^{(n+1)}=z^{(n)}-F'(y^{(n)})^{-1}F(z^{(n)}).\end{aligned}$$

Technique by Liu et al. (T_4):

$$\begin{aligned}y^{(n)}&=x^{(n)}-F'(x^{(n)})^{-1}F(x^{(n)}),\\z^{(n)}&=x^{(n)}-2\left[F'(t_1^{(n)}+t_2^{(n)})+F'(t_1^{(n)}-t_2^{(n)})\right]^{-1}F(x^{(n)}),\\x^{(n+1)}&=z^{(n)}-F'(y^{(n)})^{-1}F(z^{(n)}),\end{aligned}$$

where $t_1^{(n)}=\frac{1}{2}(y^{(n)}+x^{(n)})$ and $t_2^{(n)}=\frac{1}{2\sqrt{3}}(y^{(n)}-x^{(n)})$.

Technique by Sharma-Gupta (T_5):

$$\begin{aligned}y^{(n)}&=x^{(n)}-\frac{1}{2}F'(x^{(n)})^{-1}F(x^{(n)}),\ z^{(n)}=x^{(n)}-F'(y^{(n)})^{-1}F(x^{(n)}),\\x^{(n+1)}&=z^{(n)}-\big[2F'(y^{(n)})^{-1}-F'(x^{(n)})^{-1}\big]F(z^{(n)}).\end{aligned}$$

Technique by Solaiman-Hashim (T_6):

$$\begin{aligned}y^{(n)}&=x^{(n)}-\frac{1}{2}F'(x^{(n)})^{-1}F(x^{(n)}),\ z^{(n)}=x^{(n)}-F'(y^{(n)})^{-1}F(x^{(n)}),\\x^{(n+1)}&=z^{(n)}-\big[2F'(y^{(n)})-F'(x^{(n)})\big]^{-1}F(z^{(n)}).\end{aligned}$$

Technique by Xiao-Yin (T_7):

$$\begin{aligned}y^{(n)}&=x^{(n)}-\frac{2}{3}F'(x^{(n)})^{-1}F(x^{(n)}),\\z^{(n)}&=x^{(n)}-4\big[F'(x^{(n)})+3F'(y^{(n)})\big]^{-1}F(x^{(n)}),\\x^{(n+1)}&=z^{(n)}-\big[8(F'(x^{(n)})+3F'(y^{(n)}))^{-1}-F'(x^{(n)})^{-1}\big]F(z^{(n)}).\end{aligned}$$

Technique by Xiao-Yin (T_8):

$$\begin{aligned}y^{(n)}&=x^{(n)}+F'(x^{(n)})^{-1}F(x^{(n)}),\ z^{(n)}=y^{(n)}-F'(x^{(n)})^{-1}F(y^{(n)}),\\x^{(n+1)}&=z^{(n)}+\left[F'(y^{(n)})^{-1}-2F'(x^{(n)})^{-1}\right]F(z^{(n)}).\end{aligned}$$

Technique by Xu-Jieqing (T_9):

$$\begin{aligned} y^{(n)} &= x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}),\\ z^{(n)} &= x^{(n)} - \left[\frac{1}{8}F'(x^{(n)}) + \frac{3}{8}F'(t_1^{(n)}) + \frac{3}{8}F'(t_2^{(n)}) + \frac{1}{8}F'(y^{(n)})\right]^{-1} F(x^{(n)}),\\ x^{(n+1)} &= z^{(n)} - F'(y^{(n)})^{-1}F(z^{(n)}), \end{aligned}$$

where $t_1^{(n)} = \frac{1}{3}(2x^{(n)} + y^{(n)})$ and $t_2^{(n)} = \frac{1}{3}(x^{(n)} + 2y^{(n)})$.
Technique by Zhanlav et al. (T_{10}):

$$\begin{aligned} y^{(n)} &= x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}),\\ z^{(n)} &= y^{(n)} - (2I - F'(x^{(n)})^{-1}F'(y^{(n)}))F'(x^{(n)})^{-1}F(y^{(n)}),\\ x^{(n+1)} &= z^{(n)} - F'(x^{(n)})^{-1}F(z^{(n)}). \end{aligned}$$

Denoting the computational costs and efficiency indices, respectively by C_i and E_i, $i = 1, 2, \ldots, 10$, and then taking into consideration the computations described above, the computational costs, and the corresponding efficiency indices are listed as follows (here $D = \log d$.):

$$\begin{aligned}
C_1 &= 3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m)), \quad E_1 = \frac{1}{D}\frac{\log 5}{C_1},\\
C_2 &= 3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m)), \quad E_2 = \frac{1}{D}\frac{\log 5}{C_2},\\
C_3 &= 2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 3m - 3 + 3l(1+m)), \quad E_3 = \frac{1}{D}\frac{\log 5}{C_3},\\
C_4 &= 2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 3m + 1 + 3l(1+m)), \quad E_4 = \frac{1}{D}\frac{\log 5}{C_4},\\
C_5 &= 2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 5 + 3l(3+m)), \quad E_5 = \frac{1}{D}\frac{\log 5}{C_5},\\
C_6 &= 2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 5m - 3 + 3l(1+m)), \quad E_6 = \frac{1}{D}\frac{\log 5}{C_6},\\
C_7 &= 2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 12m - 2 + 3l(3+m)), \quad E_7 = \frac{1}{D}\frac{\log 5}{C_7},\\
C_8 &= 3m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 8 + 3l(3+m)), \quad E_8 = \frac{1}{D}\frac{\log 5}{C_8},\\
C_9 &= 2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 5m + 5 + 3l(1+m)), \quad E_9 = \frac{1}{D}\frac{\log 5}{C_9},\\
C_{10} &= 3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m)), \quad E_{10} = \frac{1}{D}\frac{\log 5}{C_{10}}.
\end{aligned}$$

3.1 Comparison of efficiencies

Consider a ratio $R_{i,j}$, which is defined below, to compare the efficiencies of iterative techniques, say T_i versus T_j,

$$R_{i,j} = \frac{E_i}{E_j} = \frac{C_j \log(r_i)}{C_i \log(r_j)},$$

where r_i and r_j, respectively, are the convergence orders of the techniques T_i and T_j. Apparently, technique T_i is computationally more efficient than T_j, only if $R_{i,j} > 1$, and we symbolize mathematically it as $T_i \gtrdot T_j$.

The proposed techniques $\boldsymbol{T_1}$ and $\boldsymbol{T_2}$, shall be analyzed and compared by analytical as well as visual approach with the existing techniques, T_i, $i = 3, 4, \ldots$, which are already expressed in this section. The efficiency indices are compared analytically by resolving the inequality $R_{i,j} > 1$, but we omit the details of verifying the results, as they are fairly straightforward. The results so obtained in analytic way are projected geometrically by presenting the boundary lines $R_{i,j} = 1$ in (ν_0, ν_1)-plane, corresponding to the particular cases of $m = 5,\ 10,\ 25,$ and 50, and taking $l = 3$ in each case. Observe that each boundary line will divide the region into two parts, where $T_i \gtrdot T_j$ on one side and $T_j \gtrdot T_i$ on the other.

In view of the above discussion, we now prove the comparison results through the following theorem:

Theorem 2. *For all $\nu_0 > 0$, $\nu_1 > 0$, and $l > 1$, we have that:*

(i) $E_1 > E_2$, *for all* $m \in \mathbb{N}$;

(ii) $E_1 > E_3$, *for* $\nu_0 < \frac{1}{3}(2m^2 - 9m + 4 + 3l(m-2))$;

(iii) $E_1 > E_4$, *for* $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 9m + 10 + 3l(m-2))$;

(iv) $E_1 > E_5$, *for* $\nu_0 < \frac{1}{6}(2m^2 - 9m + 7 + 3l(m-1))$;

(v) $E_1 > E_6$, *for* $\nu_0 < \frac{1}{3}(2m^2 - 6m + 4 + 3l(m-2))$;

(vi) $E_1 > E_7$, *for* $\nu_0 < \frac{1}{6}(2m^2 - 3m + 13 + 3l(m-1))$;

(vii) $E_1 > E_8$, *for* $m \geq 4$, *but otherwise comparison depends on the value of* l;

(viii) $E_1 > E_9$, *for* $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 6m + 16 + 3l(m-2))$;

(ix) $E_1 = E_{10}$, *for all* $m \in \mathbb{N}$;

(x) $E_2 > E_3$, *for* $\nu_0 < \frac{1}{3}(2m^2 - 9m + 1 + 3l(m-2))$;

(xi) $E_2 > E_4$, *for* $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 9m + 7 + 3l(m-2))$;

(xii) $E_2 > E_5$, *for* $\nu_0 < \frac{1}{6}(2m^2 - 9m + 1 + 3l(m-1))$;

(xiii) $E_2 > E_6$, *for* $\nu_0 < \frac{1}{3}(2m^2 - 6m + 1 + 3l(m-2))$;

(xiv) $E_2 > E_7$, *for* $\nu_0 < \frac{1}{6}(2m^2 - 3m + 7 + 3l(m-1))$;

(xv) $E_2 > E_8$, *for* $m \geq 4$, *but otherwise comparison depends on the value of* l;

(xvi) $E_2 > E_9$, *for* $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 6m + 13 + 3l(m-2))$;

(xvii) $E_2 < E_{10}$, *for all* $m \in \mathbb{N}$.

Proof. $\boldsymbol{T_1}$ *versus* $\boldsymbol{T_2}$ *case*: In this case, the ratio is

$$R_{1,2} = \frac{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

It can be easily verified that the inequality $R_{1,2} > 1$ holds for all $m \in \mathbb{N}$, and consequently $\boldsymbol{T_1} \gtrdot \boldsymbol{T_2}$ for all $m \in \mathbb{N}$.

$\boldsymbol{T_1}$ *versus* T_3 *case*: In this case, the ratio is

$$R_{1,3} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 3m - 3 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

The inequality $R_{1,3} > 1$ resolves into $\nu_0 < \frac{1}{3}(2m^2 - 9m + 4 + 3l(m-2))$, which eventually proves (ii). The boundary lines $R_{1,3} = 1$, in (ν_0, ν_1)-plane, are presented in Figure 1, where $\boldsymbol{T_1} \gtrdot T_3$ in the region which is below the horizontal line for each particular case of m.

$\boldsymbol{T_1}$ *versus* T_4 *case*: In this case, the ratio is

$$R_{1,4} = \frac{2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 3m + 1 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

The inequality $R_{1,4} > 1$ simply results into $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 9m + 10 + 3l(m-2))$, and this concludes (iii). The boundary lines in (ν_0, ν_1)-plane, with positive slopes, are shown in Figure 2, where $\boldsymbol{T_1} \gtrdot T_4$ to the right (below) side of line for each value of m.

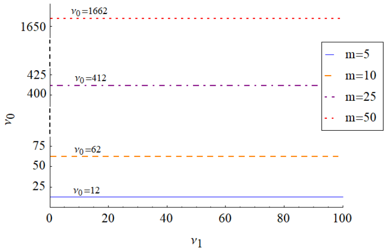

Figure 1. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_3.

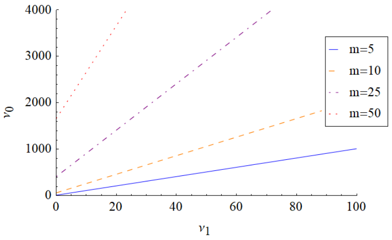

Figure 2. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_4.

$\boldsymbol{T_1}$ *versus* T_5 *case*: In this case, the ratio is

$$R_{1,5} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 5 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

By resolving the inequality $R_{1,5} > 1$, it is deduced that $\nu_0 < \frac{1}{6}(2m^2 - 9m + 7 + 3l(m-1))$, which ultimately proves (iv). In this case, the boundary lines are presented in Figure 3 with $\boldsymbol{T_1} \gtrdot T_5$ on the lower side of line for each case of m.

$\boldsymbol{T_1}$ *versus* T_6 *case*: In this case, the ratio is

$$R_{1,6} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 5m - 3 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

The resolution of inequality $R_{1,6} > 1$ results into $\nu_0 < \frac{1}{3}(2m^2 - 6m + 4 + 3l(m-2))$, and this proves (v). The boundary lines, for comparison, are depicted in Figure 4, where $\boldsymbol{T_1} \gtrdot T_6$ holds in the lower region of line for each particular case.

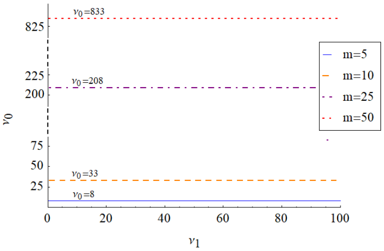

Figure 3. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_5.

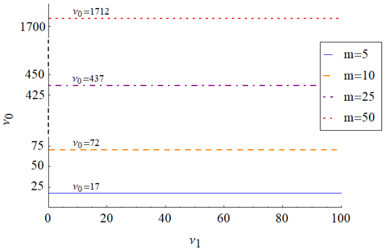

Figure 4. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_6.

$\boldsymbol{T_1}$ *versus* T_7 *case*: In this case, the ratio is

$$R_{1,7} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 12m - 2 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

The inequality $R_{1,7} > 1$ resolves into $\nu_0 < \frac{1}{6}(2m^2 - 3m + 13 + 3l(m-1))$, which concludes (vi). The boundary lines in this case are shown in Figure 5, with $\boldsymbol{T_1} \gtrdot T_7$ on the lower side of each line.

$\boldsymbol{T_1}$ *versus* T_8 *case*: In this case, the ratio is

$$R_{1,8} = \frac{3m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 8 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

It is straightforward to verify that, for $\nu_0, \nu_1 > 0$, and $l > 1$, the inequality $R_{1,8} > 1$ holds for all $m \geq 4$, but otherwise for $2 \leq m < 4$, the inequality holds, provided $l > \frac{2m^2 - 9m + 1}{3 - 3m}$. This proves (vii), and hence $\boldsymbol{T_1} \gtrdot T_8$ for all $m \geq 4$, but otherwise comparison depends on the parameter l.

$\boldsymbol{T_1}$ *versus* T_9 *case*: In this case, the ratio is

$$R_{1,9} = \frac{2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 5m + 5 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}.$$

Resolution of the inequality $R_{1,9} > 1$ results into $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 6m + 16 + 3l(m-2))$, and this proves (viii). The boundary lines for the comparison are shown in Figure 6 with $\boldsymbol{T_1} \gtrdot T_9$ on the lower (right) side of line for each case.

$\boldsymbol{T_1}$ *versus* T_{10} *case*: It is clear that for this case, the ratio is

$$R_{1,10} = 1,$$

which immediately concludes that $E_1 = E_{10}$ for all $m \in \mathbb{N}$.

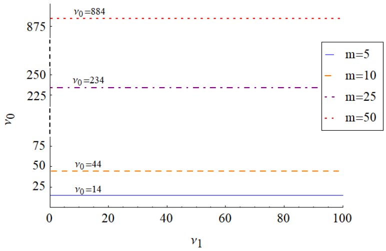

Figure 5. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_7.

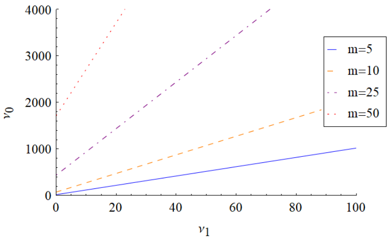

Figure 6. Boundary lines for comparison of $\boldsymbol{T_1}$ and T_9.

$\boldsymbol{T_2}$ *versus* T_3 *case*: In this case, the ratio is

$$R_{2,3} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 3m - 3 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

The inequality $R_{2,3} > 1$ simply resolves into the relation $\nu_0 < \frac{1}{3}(2m^2 - 9m + 1 + 3l(m-2))$, which ultimately proves (x). The boundary lines in (ν_0, ν_1)-plane are depicted in Figure 7 with $\boldsymbol{T_2} \gtrdot T_3$ on the lower side of line for each value of m.

$\boldsymbol{T_2}$ *versus* T_4 *case*: In this case, the ratio is

$$R_{2,4} = \frac{2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 3m + 1 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

By resolving the inequality $R_{2,4} > 1$, it is deduced that $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 9m + 7 + 3l(m-2))$, which concludes (xi). The boundary lines with positive slopes are shown in Figure 8, where $\boldsymbol{T_2} \gtrdot T_4$ on the right (below) side of line for each particular case of m.

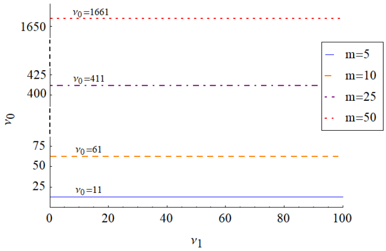

Figure 7. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_3.

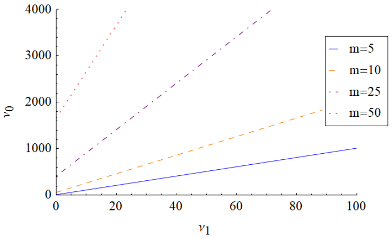

Figure 8. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_4.

$\boldsymbol{T_2}$ *versus* T_5 *case*: In this case, the ratio is

$$R_{2,5} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 5 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

The inequality $R_{2,5} > 1$ simply results into $\nu_0 < \frac{1}{6}(2m^2 - 9m + 1 + 3l(m-1))$, and this proves (xii). The boundary lines for comparison are presented in Figure 9 with $\boldsymbol{T_2} \gtrdot T_5$ on the lower side of line for each case.

$\boldsymbol{T_2}$ *versus* T_6 *case*: In this case, the ratio is

$$R_{2,6} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{2}(2m^2 + 5m - 3 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

The resolution of inequality $R_{2,6} > 1$ results into $\nu_0 < \frac{1}{3}(2m^2 - 6m + 1 + 3l(m-2))$, which ultimately proves (xiii). The boundary lines are shown in Figure 10, where $\boldsymbol{T_2} \gtrdot T_6$ on the lower side of each line.

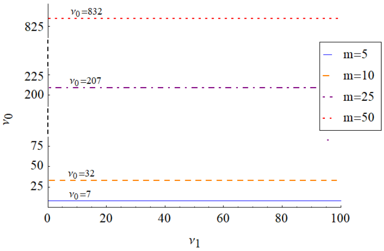

Figure 9. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_5.

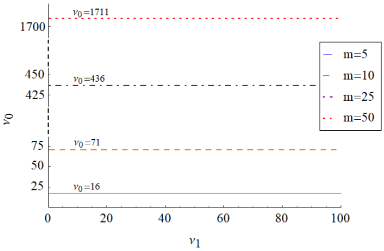

Figure 10. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_6.

$\boldsymbol{T_2}$ *versus* T_7 *case*: In this case, the ratio is

$$R_{2,7} = \frac{2m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 12m - 2 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

By resolving the inequality $R_{2,7} > 1$, it is simple to deduce that $\nu_0 < \frac{1}{6}(2m^2 - 3m + 7 + 3l(m-1))$, which proves (xiv). The boundary lines in (ν_0, ν_1)-plane are depicted in Figure 11 with $\boldsymbol{T_2} \gtrdot T_7$ on the lower side of line for each case of m.

$\boldsymbol{T_2}$ *versus* T_8 *case*: In this case, the ratio is

$$R_{2,8} = \frac{3m\nu_0 + 2m^2\nu_1 + \frac{m}{3}(2m^2 + 9m - 8 + 3l(3+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

Given $\nu_0, \nu_1 > 0$ and $l > 1$, it is simple to verify that, for all $m \geq 4$, the inequality $R_{2,8} > 1$ holds, but for $2 \leq m < 4$, the inequality holds only when $l > \frac{2m^2-9m-5}{3-3m}$. This proves (xv), and hence $\boldsymbol{T_2} \gtrdot T_8$ for all $m \geq 4$, but otherwise comparison depends on the value of l.

$\boldsymbol{T_2}$ *versus* T_9 *case*: In this case, the ratio is

$$R_{2,9} = \frac{2m\nu_0 + 4m^2\nu_1 + \frac{m}{2}(2m^2 + 5m + 5 + 3l(1+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

The inequality $R_{2,9} > 1$ simply resolves into $\nu_0 < 2m\nu_1 + \frac{1}{3}(2m^2 - 6m + 13 + 3l(m-2))$, and this concludes (xvi). The boundary lines for comparison are shown in Figure 12 with $\boldsymbol{T_2} \gtrdot T_9$ on the right (below) side of each line.

$\boldsymbol{T_2}$ *versus* T_{10} *case*: In this case, the ratio is

$$R_{2,10} = \frac{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 17 + 3l(7+m))}{3m\nu_0 + 2m^2\nu_1 + \frac{m}{6}(2m^2 + 27m - 11 + 3l(7+m))}.$$

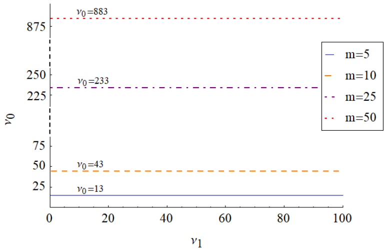

Figure 11. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_7.

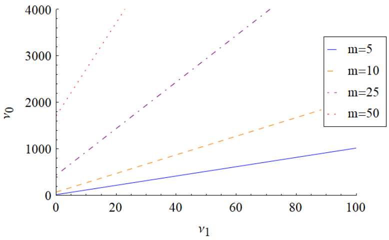

Figure 12. Boundary lines for comparison of $\boldsymbol{T_2}$ and T_9.

It can be clearly observed that the inequality $R_{2,10} < 1$ holds for all $m \in \mathbb{N}$, and consequently $T_{10} \gtrdot \boldsymbol{T_2}$ for all $m \in \mathbb{N}$.

□

It can be deduced from the above analysis that both of the proposed techniques display better efficiency in comparison to the existing techniques with increasing values of m. We conclude this section with a remark that, as large as the system is, the proposed techniques are, in general, superior than the existing ones in reference to the subject of computational complexity.

4 Numerical experimentation

The numerical experimentation shall be carried out to assess the performance and stability of the proposed techniques by executing their algorithms on a digital platform. The outcome of testing needs to be compared with the corresponding outcome of existing techniques to arrive at some valid and logical conclusion. Some of the nonlinear problems, emerging from different practical situations, have been selected for this purpose. The performance of an iterative technique is generally evaluated on the basis of two factors: (i) Number of iterations required to converge, and (ii) CPU time elapsed during the entire course of action. Both of these factors tend to vary with the convergence behavior of technique as well as with the proximity of initial approximation. Let us note that the elapsed CPU time also varies with the characteristics of digital platform. In our case, numerical experimentation is being done using the multi-precision arithmetic software *Mathematica* [12], which is installed on the machine with specifications: Intel(R) Core (TM) i5-9300H processor and Windows 10 operating system.

Numerical performance of an iterative technique is considerably governed by the number of evaluations or mathematical operations involved per iteration. This indicates the existence of correlation between the performance and efficiency. To build this relation, the evaluation cost of each mathematical operation and function needs to be expressed in terms of product units. As discussed in Section 3, the numerical estimates of parameters ν_0, ν_1, and l are required for that purpose. In this regard, Table 1 displays the CPU time (in milliseconds) elapsed during the execution of elementary operations and functions, and their estimation of cost in units of product. Notice that the cost of division is approximately thrice the cost of product.

Table 1. CPU time and estimation of computational cost of elementary functions, where $x = \sqrt{3} - 1$ and $y = \sqrt{5}$ (with 4096 digits of accuracy).

Functions	$x * y$	x/y	$\sqrt{x}$	e^x	$\log(x)$	$\sin(x)$	$\cos(x)$	$\arctan(x)$
CPU Time	0.0172	0.0484	0.0234	1.5562	1.3469	1.6938	1.6896	2.9797
Cost	1	2.81	1.36	90.48	78.31	98.48	98.23	173.24

To demonstrate the performance of proposed techniques, and to further compare these with the existing ones, their algorithms are executed using the software *Mathematica* under the similar framework. The following stopping criterion is used to terminate the iterations:

$$\|x^{(n)} - x^{(n-1)}\| + \|F(x^{(n)})\| < 10^{-100}.$$

Further, the approximated computational order of convergence (ACOC) is computed to authenticate the theoretically established convergence order, which is given by the expression (see [2]),

$$\text{ACOC} = \frac{\ln\left(\|x^{(n)} - x^{(n-1)}\| / \|x^{(n-1)} - x^{(n-2)}\|\right)}{\ln\left(\|x^{(n-1)} - x^{(n-2)}\| / \|x^{(n-2)} - x^{(n-3)}\|\right)}.$$

Now, considering the following nonlinear problems for the performance analysis, we display the outcomes in respect of: (i) Number of iterations (n), (ii) ACOC, (iii) Computational cost (C_i), (iv) Efficiency index (E_i), and (v) Elapsed CPU time (in seconds). Note that, to illustrate the efficiency index of techniques, we conveniently choose $D = 10^{-5}$ for each problem.

Problem 1. Starting with the three dimensional nonlinear problem, which is given by

$$\begin{aligned} x^2 + y^2 + z^2 &= 1, \\ 2x^2 + y^2 + 4z &= 0, \\ 3x^2 - 4y^2 + z^2 &= 0, \end{aligned}$$

the initial approximation is taken as $\left(-\frac{3}{2}, -\frac{3}{2}, -\frac{3}{2}\right)^T$ to obtain the particular solution $x^* = (-0.6982..., -0.6285..., -0.3425...)^T$.

To evaluate the computational cost and efficiency index for this problem, the parameters used in equation (3.2) are estimated as, $(m, \nu_0, \nu_1, l) = (3, 2.33, 0.67, 2.81)$. Numerical results for the performance of methods are displayed in the Table 2.

Problem 2. Consider the boundary value problem (see [2]), investigated in the study of finite deflections of an elastic string under the transverse load, as follows:

$$y''(t) + a^2(y'(t))^2 + 1 = 0, \quad y(0) = 0, \quad y(1) = 0. \tag{4.1}$$

Table 2. Comparison of performance of techniques for Problem 1.

Technique	n	ACOC	C_i	E_i	CPU Time
$\boldsymbol{T_1}$	4	4.984	116.15	1385.65	0.0323
$\boldsymbol{T_2}$	4	4.999	119.15	1350.77	0.0342
T_3	4	4.993	112.58	1429.59	0.0624
T_4	4	4.993	130.58	1232.53	0.0626
T_5	4	4.989	116.58	1380.54	0.0478
T_6	4	4.993	121.58	1323.77	0.0632
T_7	4	4.989	128.58	1251.70	0.0637
T_8	4	4.982	120.58	1334.75	0.0493
T_9	4	4.993	145.58	1105.54	0.0784
T_{10}	4	4.984	116.15	1385.65	0.0331

The exact solution of above equation is $y(t) = \frac{1}{a^2}\ln\left(\frac{\cos(a(t-1/2))}{\cos(a/2)}\right)$. We intend to transform the given problem into a finite dimensional problem by considering the uniform partition of $[0,1]$, with sub-intervals of length $h = 1/k$, as

$$0 = t_0 < t_1 < t_2 < \cdots < t_{k-1} < t_k = 1, \text{ where } t_i{=}t_0 + ih, \ (i{=}1,2,\ldots,k-1).$$

Denoting $y(t_i) = y_i$ for each $i = 1,2,\ldots,k-1$, and approximating the derivatives in (4.1) by the second order divided differences,

$$y_i' = \frac{y_{i+1} - y_{i-1}}{2h}, \quad y_i'' = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2},$$

we obtain the system of nonlinear equations in $k-1$ variables:

$$y_{i-1} - 2y_i + y_{i+1} + \frac{a^2}{4}(y_{i+1} - y_{i-1})^2 + h^2 = 0, \quad (i = 1,2,\ldots,k-1).$$

In particular for $a = 2$, and setting $k = 16$, the given system reduces to 15 nonlinear equations satisfying the solution,

$$x^* = (0.0426..,0.0755..,0.1012..,0.1209..,0.1355..,0.1455..,0.1515..,0.1534..,\\ 0.1515..,0.1455..,0.1355..,0.1209..,0.1012..,0.0755..,0.0426..)^T.$$

The initial estimate to the given solution is chosen as $(-\frac{3}{2},\overset{15}{\cdots\cdots},-\frac{3}{2})^T$. The approximate numerical solution, so obtained, is compared with the exact solution of the given problem in the Figure 13.

In this problem, the estimated values of parameters, used in the equation (3.2), are given by $(m,\nu_0,\nu_1,l) = (15,2,0.067,2.81)$. Table 3 exhibits the comparison of performance of techniques.

Problem 3. Consider the nonlinear integral equation (see [1]),

$$u(s) = \frac{7}{8}s + \frac{1}{2}\int_0^1 s\ t\ u(t)^2 dt, \tag{4.2}$$

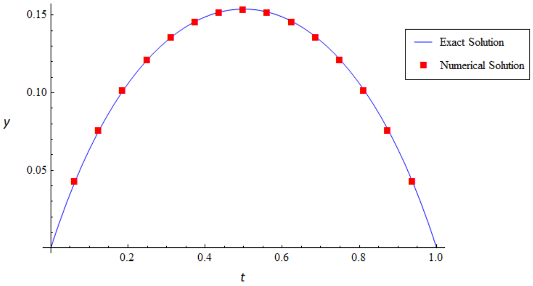

Figure 13. Graphical comparison of exact and numerical solution of Problem 2.

Table 3. Comparison of performance of techniques for Problem 2.

Technique	n	ACOC	C_i	E_i	CPU Time
$\boldsymbol{T_1}$	5	5.000	2678.65	60.08	0.2664
$\boldsymbol{T_2}$	4	5.000	2693.65	59.75	0.2963
T_3	5	5.000	4791.60	33.59	0.5472
T_4	5	5.000	4851.60	33.17	0.5945
T_5	5	5.000	3748.70	42.93	0.4373
T_6	5	5.000	5016.60	32.08	0.6102
T_7	5	5.000	3988.70	40.35	0.4532
T_8	5	5.000	3763.70	42.76	0.4477
T_9	5	5.000	5106.60	31.52	0.5934
T_{10}	5	5.000	2678.65	60.08	0.2682

where $s, t \in [0,1]$, and $u \in C[0,1]$, with $C[0,1]$ being a space of continuous functions on the unit interval.

To transform the Equation (4.2) into a finite-dimensional problem, the given interval $[0,1]$ is partitioned into sub-intervals of equal length, $h = 1/k$, as follows:

$$0 = t_0 < t_1 < t_2 < \cdots < t_{k-1} < t_k = 1, \text{ where } t_i = t_0 + ih, \quad (i = 1, 2, \ldots, k-1).$$

Denoting $u(t_i) = u_i$ for each $i = 1, 2, \ldots, k$, and approximating the integral in Equation (4.2) using the trapezoidal rule of integration, we obtain the system of k nonlinear equations as

$$\frac{7}{8}s_i - u_i + \frac{hs_i}{2}\left(\frac{1}{2}u_k^2 + \sum_{j=1}^{k-1} t_j u_j^2\right) = 0, \quad (i = 1, 2, \ldots, k), \tag{4.3}$$

where $s_i = t_i = i/k$ for each i.

Setting $k = 25$ in particular, the solution of system of equations (4.3) is given as

$$x^* = (0.04001.., 0.08002.., 0.12003.., 0.16004.., 0.20005.., 0.24006.., 0.28007.., 0.32008.., 0.36009.., 0.40010.., 0.44011.., 0.48012.., 0.52013.., 0.56014.., 0.60016.., 0.64017.., 0.68018.., 0.72019.., 0.76020.., 0.80021.., 0.84022.., 0.88023.., 0.92024.., 0.96025.., 1.00026..)^T.$$

We set the initial approximation as $(-1, \overset{25}{\cdots\cdots}, -1)^T$. Numerical results so obtained are displayed in Table 4.

Table 4. Comparison of performance of techniques for Problem 3.

Technique	n	ACOC	C_i	E_i	CPU Time
$\boldsymbol{T_1}$	4	5.000	10549.00	15.26	1.1062
$\boldsymbol{T_2}$	4	5.000	10574.00	15.22	1.1095
T_3	4	5.000	20664.75	07.79	2.0311
T_4	4	5.000	21964.75	07.33	2.1405
T_5	4	5.000	15617.00	10.31	1.6112
T_6	4	5.000	21289.75	07.56	2.1254
T_7	4	5.000	16267.00	09.89	1.6256
T_8	4	5.000	15667.00	10.27	1.6245
T_9	4	5.000	22639.75	07.11	2.1091
T_{10}	4	5.000	10549.00	15.26	1.1078

The exact solution of the given integral equation along with numerical solution obtained is compared in the Figure 14. Moreover, the estimated values of the parameters are, $(m, \nu_0, \nu_1, l) = (25, 3, 1, 2.81)$.

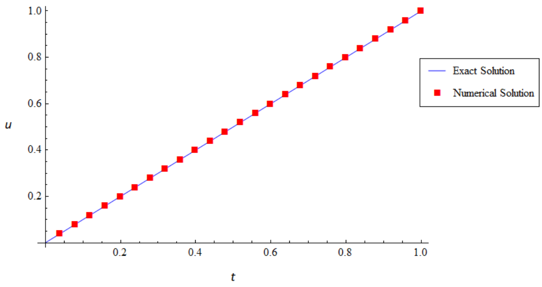

Figure 14. Graphical comparison of exact and numerical solution of Problem 3.

Problem 4. Now let us take a large system of equations as,

$$e^{-x_i} - \sum_{j=1, j\neq i}^{m} x_j = 0, \quad (i = 1, 2, \ldots, m),$$

where $m = 200$. We set the initial estimate as $\left(\frac{3}{2}, \cdots, \frac{3}{2}\right)^T$ to obtain the solution,

$$x^* = (0.0050\ldots, \cdots, 0.0050\ldots)^T.$$

In this problem, the values of parameters are estimated as $(m, \nu_0, \nu_1, l) = (200, 90.48, 0.45, 2.81)$. Table 5 depicts the numerical results of performance for the considered problem.

Table 5. Comparison of performance of techniques for Problem 4.

Technique	n	ACOC	C_i	E_i	CPU Time
$\boldsymbol{T_1}$	3	5.000	2.99E+06	5.37E-02	60.750
$\boldsymbol{T_2}$	3	5.000	2.99E+06	5.37E-02	60.469
T_3	3	5.000	8.30E+06	1.94E-02	175.813
T_4	3	5.000	8.34E+06	1.93E-02	183.766
T_5	3	5.000	5.64E+06	2.85E-02	154.375
T_6	3	5.000	8.34E+06	1.93E-02	182.187
T_7	3	5.000	5.68E+06	2.83E-02	142.969
T_8	3	5.000	5.66E+06	2.84E-02	128.641
T_9	3	5.000	8.38E+06	1.92E-02	175.407
T_{10}	3	5.000	2.99E+06	5.37E-02	60.653

Problem 5. At last, we again consider a large system of equations as,

$$\begin{aligned} &x_i + \log(2 + x_i + x_{i+1}) = 0, \quad (i = 1, 2, \ldots, m-1), \\ &x_m + \log(2 + x_m + x_1) = 0. \end{aligned}$$

By setting $m = 500$, the initial estimate is selected as $\left(-\frac{1}{4}, \cdots, -\frac{1}{4}\right)^T$ for the solution,

$$x^* = (-0.3149\ldots, \cdots, -0.3149\ldots)^T.$$

The estimated values of parameters for this problem are, $(m, \nu_0, \nu_1, l) = (500, 78.31, 0.0056, 2.81)$. The numerical performance is displayed in the Table 6.

It can be inferred from the findings of performance outcomes, displayed in Tables 2–6, that the proposed fifth order techniques are computationally more efficient as compared to their existing counterparts, as both of these exhibit superiority in terms of elapsed CPU time as well as efficiency index. Computation of ACOC further authenticates the theoretically proven fifth order of convergence. Similar numerical tests, conducted for a variety of problems, largely confirm the above conclusions.

Table 6. Comparison of performance of techniques for Problem 5.

Technique	n	ACOC	C_i	E_i	CPU Time
$\boldsymbol{T_1}$	3	5.000	4.33E+07	3.72E-03	17.593
$\boldsymbol{T_2}$	3	5.000	4.33E+07	3.72E-03	17.675
T_3	3	5.000	1.27E+08	1.27E-03	24.515
T_4	3	5.000	1.27E+08	1.27E-03	24.343
T_5	3	5.000	8.49E+07	1.90E-03	19.109
T_6	3	5.000	1.27E+08	1.27E-03	24.157
T_7	3	5.000	8.51E+07	1.89E-03	19.375
T_8	3	5.000	8.49E+07	1.90E-03	21.875
T_9	3	5.000	1.27E+08	1.27E-03	24.969
T_{10}	3	5.000	4.33E+07	3.72E-03	17.611

5 Conclusions

Two multi-step iterative techniques, for solving the systems of nonlinear equations, have been proposed in the present study. The techniques are found to possess the fifth order of convergence under some prescribed assumptions. With the fact that only single evaluation of matrix inversion is required per iteration, both of the developed techniques are highly economical from the view of computational complexity, typically for the large scale systems. Iterative techniques with these characteristics are hardly found in the literature. Performance and stability of the proposed techniques have been assessed by executing the numerical experimentation on the selected nonlinear problems arising in different practical situations. Outcomes of the testing conclude that the proposed techniques dominate the majority of existing ones when examined in the context of computational cost, efficiency index, convergence behavior, and the elapsed CPU time in the execution of algorithm.

References

[1] Z. Avazzadeh, M. Heydari and G.B. Loghmani. Numerical solution of Fredholm integral equations of the second kind by using integral mean value theorem. *Applied Mathematical Modelling*, **35**(5):2374–2383, 2011. https://doi.org/10.1016/j.apm.2010.11.056.

[2] A. Cordero, J.L. Hueso, E. Martínez and J.R. Torregrosa. Efficient high-order methods based on golden ratio for nonlinear systems. *Applied Mathematics and Computation*, **217**(9):4548–4556, 2011. https://doi.org/10.1016/j.amc.2010.11.006.

[3] A. Cordero, J.L. Hueso, E. Martínez and J.R. Torregrosa. Increasing the convergence order of an iterative method for nonlinear systems. *Applied Mathematics Letters*, **25**(12):2369–2374, 2012. https://doi.org/10.1016/j.aml.2012.07.005.

[4] Z. Liu, Q. Zheng and C.E. Huang. Third-and fifth-order Newton–Gauss methods for solving nonlinear equations with n variables. *Applied Mathematics and Computation*, **290**:250–257, 2016. https://doi.org/10.1016/j.amc.2016.06.010.

[5] J.M. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, 1970.

[6] A.M. Ostrowski. *Solution of Equation and Systems of Equations.* Academic Press, New York, 1960.

[7] F.A. Potra and V. Pták. On a class of modified Newton processes. *Numerical Functional Analysis and Optimization*, **2**(1):107–120, 1980. https://doi.org/10.1080/01630568008816049.

[8] J.R. Sharma and P. Gupta. An efficient fifth order method for solving systems of nonlinear equations. *Computers & Mathematics with Applications*, **67**(3):591–601, 2014. https://doi.org/10.1016/j.camwa.2013.12.004.

[9] R. Sihwail, O.S. Solaiman, K. Omar, K.A.Z. Ariffin, M. Alswaitti and I. Hashim. A hybrid approach for solving systems of nonlinear equations using Harris Hawks optimization and Newton's method. *IEEE Access*, **9**:95791–95807, 2021. https://doi.org/10.1109/ACCESS.2021.3094471.

[10] O.S. Solaiman and I. Hashim. An iterative scheme of arbitrary odd order and its basins of attraction for nonlinear systems. *Computers, Materials & Continua*, **66**(2):1427–1444, 2021. https://doi.org/10.32604/cmc.2020.012610.

[11] J.F. Traub. *Iterative Methods for the Solution of Equations.* Chelsea Publishing Company, New York, 1982.

[12] S. Wolfram. *The Mathematica Book (*5^{th} *edition).* Wolfram Media, USA, 2003.

[13] X.Y. Xiao and H.W. Yin. A new class of methods with higher order of convergence for solving systems of nonlinear equations. *Applied Mathematics and Computation*, **264**:300–309, 2015. https://doi.org/10.1016/j.amc.2015.04.094.

[14] X.Y. Xiao and H.W. Yin. Increasing the order of convergence for iterative methods to solve nonlinear systems. *Calcolo*, **53**(3):285–300, 2016. https://doi.org/10.1007/s10092-015-0149-9.

[15] Z. Xu and T. Jieqing. The fifth order of three-step iterative methods for solving systems of nonlinear equations. *Mathematica Numerica Sinica*, **35**(3):297–304, 2013.

[16] T. Zhanlav, C. Chun, K. Otgondorj and V. Ulziibayar. High-order iterations for systems of nonlinear equations. *International Journal of Computer Mathematics*, **97**(8):1704–1724, 2020. https://doi.org/10.1080/00207160.2019.1652739.

[17] T. Zhanlav and K. Otgondorj. Higher order Jarratt-like iterations for solving systems of nonlinear equations. *Applied Mathematics and Computation*, **395**:125849, 2021. https://doi.org/10.1016/j.amc.2020.125849.