WILEY | Hindawi

Research Article

Efficient High-Order Iterative Methods for Solving Nonlinear Systems and Their Application on Heat Conduction Problems

Alicia Cordero,[1] Esther Gómez,[2] and Juan R. Torregrosa[1]

[1]*Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, Valencia, Spain*
[2]*Universitat de València, Valencia, Spain*

Correspondence should be addressed to Alicia Cordero; acordero@mat.upv.es

For solving nonlinear systems of big size, such as those obtained by applying finite differences for approximating the solution of diffusion problem and heat conduction equations, three-step iterative methods with eighth-order local convergence are presented. The computational efficiency of the new methods is compared with those of some known ones, obtaining good conclusions, due to the particular structure of the iterative expression of the proposed methods. Numerical comparisons are made with the same existing methods, on standard nonlinear systems and a nonlinear one-dimensional heat conduction equation by transforming it in a nonlinear system by using finite differences. From these numerical examples, we confirm the theoretical results and show the performance of the presented schemes.

1. Introduction

The design of fixed point iterative methods for solving equations and systems of nonlinear equations is an important and challenging task in the field of numerical analysis. Nonlinearity is ubiquitous in physical phenomena. Fluid and plasma mechanics, gas dynamics, elasticity, relativity, chemical reactions, combustion, ecology, biomechanics, economics modeling problems, transport theory, and many other phenomena are all governed inherently by nonlinear equations. For this reason, an ever increasing proportion of modern mathematical research is devoted to the analysis of nonlinear systems and nonlinear phenomena. These and other more examples allow us to affirm that finding the solution $\bar{x}$ of a nonlinear system $F(x) = 0$ is a classical and difficult problem with many applications in sciences and engineering, wherein $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a sufficiently Fréchet differentiable function in an open convex set D. In the last years, many iterative methods have been constructed for solving nonlinear systems; see, for example, [1–6] and the references therein. The best known method for finding a solution $\bar{x} \in D$ is Newton's scheme:

$$x^{(k+1)} = x^{(k)} - \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right), \quad k = 0, 1, 2, \dots, \quad (1)$$

where $F'(x^{(k)})$ is the Jacobian matrix of function F evaluated in the kth iteration.

Based on Newton's or Newton-like iterations, some high-order methods for computing a solution of nonlinear system $F(x) = 0$ have been proposed in the literature. The aim of these new schemes is to accelerate the convergence or to improve the computational efficiency. For example, among other authors, Soleymani et al. in [7] constructed a multistep class of sixth-order iterative method for solving nonlinear systems; Hueso et al. in [4] developed sixth-order iterative methods requiring two evaluations of function F and two of Jacobian F' per iteration; Sharma and Arora in [6] designed a sixth-order method which requires three functional and two Jacobian evaluations per iteration, and Xiao and Yin [8] developed a three-step iterative scheme of order five. On the other hand, Wang et al. in [9] constructed a seventh-order derivative free iterative method by evaluating the first order divided difference operator $[x, y; F]$ three times per iteration.

Specifically, the sixth-order scheme designed by Soleymani et al. [7], that we denote by SLB, is a three-step Jarratt-type method whose iterative expression is

$$y^{(k)} = x^{(k)} - \frac{2}{3} \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right),$$

$$z^{(k)} = x^{(k)} - \frac{1}{2} \left[3F'\left(y^{(k)}\right) - F'\left(x^{(k)}\right) \right]^{-1} \left[3F'\left(y^{(k)}\right) \right.$$

$$+ F'\left(x^{(k)}\right) \left] \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right),$$

$$x^{(k+1)} = z^{(k)} - \left(\frac{1}{2} \left[3F'\left(y^{(k)}\right) - F'\left(x^{(k)}\right) \right]^{-1} \right.$$

$$\left. \cdot \left[3F'\left(y^{(k)}\right) + F'\left(x^{(k)}\right) \right] \right)^2 \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(z^{(k)}\right).$$

$$\tag{2}$$

Hueso et al. in [4] developed several iterative schemes of order six; we use one of them in the numerical section for comparing it with our proposed schemes on different test problems. In particular, in [4] the authors present the following method that we denote by HMT:

$$y^{(k)} = x^{(k)} - \frac{2}{3} \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right),$$

$$z^{(k)} = x^{(k)} - \left[\frac{5}{8} I + \frac{3}{8} \left(\left[F'\left(y^{(k)}\right) \right]^{-1} F'\left(x^{(k)}\right) \right)^2 \right]$$

$$\cdot \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right), \tag{3}$$

$$x^{(k+1)} = z^{(k)} - \left[-\frac{9}{4} I + \frac{15}{8} \left[F'\left(y^{(k)}\right) \right]^{-1} F'\left(x^{(k)}\right) \right.$$

$$\left. + \frac{11}{8} \left[F'\left(x^{(k)}\right) \right]^{-1} F'\left(y^{(k)}\right) \right] \left[F'\left(y^{(k)}\right) \right]^{-1} F\left(z^{(k)}\right),$$

where I denotes the identity matrix of size $n \times n$.

On the other hand, Xiao and Yin in [8] described the following three-step fifth-order scheme that we denote by XY:

$$y^{(k)} = x^{(k)} + \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right),$$

$$z^{(k)} = y^{(k)} - \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(y^{(k)}\right),$$

$$x^{(k+1)} = z^{(k)} \tag{4}$$

$$+ \left(\left[F'\left(y^{(k)}\right) \right]^{-1} - 2 \left[F'\left(x^{(k)}\right) \right]^{-1} \right) F\left(z^{(k)}\right).$$

In order to compare the different methods under the point of view of the computational cost, Ostrowski in [10] defined the efficiency index as $I = p^{1/d}$, where p is the order of convergence and d is the number of functional evaluations per iteration. Let us remark that for evaluating function F we need n scalar functional evaluations (the coordinate functions of F), whilst for evaluating Jacobian F' it is necessary to evaluate n^2 functions (all the entries of matrix F'). On the other hand, all the iterative methods for solving nonlinear systems require one or more matrix inversion; that is, one or more linear systems must be solved. So, the number of operations needed for solving a linear system plays in this context an important role. For this reason, the authors introduced in [2] the computational efficiency index, CI, which combines the efficiency index defined by Ostrowski and the number of products-quotients required

per iteration. We define this index as CI $= p^{1/(d+\text{op})}$, where op is the number of products-quotients per iteration.

We recall that the number of products and quotients required for solving a linear system by Gaussian elimination is $(1/3)n^3 + n^2 - (1/3)n$, where n is the size of the system. In addition, for solving q linear systems with the same matrix of coefficients, by using LU decomposition, we need $(1/3)n^3 + qn^2 - (1/3)n$ products-quotients. By using this information, in Section 3, we compare the computational efficiency indices of the different methods used in this manuscript.

The main objective of this paper is to develop high-order iterative methods in such a way that they involve as lower computational cost as possible and they have good stability properties on big-sized systems. From Newton's method and by using variation parameters procedure we present a two-step iterative method of order five. This order can be increased in three units by adding a new step with the same structure as the second one. This idea is generalized for obtaining an iterative method of arbitrary order increasing in three units the order each time that we add a new step with the same structure as the previous one. In each new step we only need a new functional evaluation.

In order to analyze the convergence properties of the different new schemes that will be introduced in this paper, we need to recall several concepts and tools, some of them introduced by the authors in [2].

1.1. Basic Definitions. Let $\{x^{(k)}\}_{k \geq 0}$ be a sequence in $\mathbb{R}^n$ which converges to $\overline{x}$. Then, convergence is called of order p, $p \geq 1$, if there exists $M > 0$ $(0 < M < 1$ if $p = 1)$ and k_0 such that

$$\left\| x^{(k+1)} - \overline{x} \right\| \leq M \left\| x^{(k)} - \overline{x} \right\|^p, \quad \forall k \geq k_0, \tag{5}$$

or

$$\left\| e^{(k+1)} \right\| \leq M \left\| e^{(k)} \right\|^p, \quad \forall k \geq k_0, \tag{6}$$

where $e^{(k)} = x^{(k)} - \overline{x}$.

The following notation was introduced in [2], but we present it for completeness. Let $F : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be sufficiently Fréchet differentiable in D. The qth derivative of F at $u \in \mathbb{R}^n$, $q \geq 1$, is the q-linear function $F^{(q)}(u) : \mathbb{R}^n \times \cdots \times \mathbb{R}^n \to \mathbb{R}^n$ such that $F^{(q)}(u)(v_1, \ldots, v_q) \in \mathbb{R}^n$. It is easy to observe the following:

(1) $F^{(q)}(u)(v_1, \ldots, v_{q-1}, \cdot) \in \mathscr{L}(\mathbb{R}^n)$.

(2) $F^{(q)}(u)(v_{\sigma(1)}, \ldots, v_{\sigma(q)}) = F^{(q)}(u)(v_1, \ldots, v_q)$, for all permutation σ of $\{1, 2, \ldots, q\}$.

From the above properties we can use the following notation:

(a) $F^{(q)}(u)(v_1, \ldots, v_q) = F^{(q)}(u)v_1 \ldots v_q$.

(b) $F^{(q)}(u)v^{q-1}F^{(p)}v^p = F^{(q)}(u)F^{(p)}(u)v^{q+p-1}$.

On the other hand, for $\xi + h \in \mathbb{R}^n$ lying in a neighborhood of a solution $\overline{x}$ of $F(x) = 0$, we can apply Taylor's expansion

and, assuming that the Jacobian matrix $F'(\overline{x})$ is nonsingular, we have

$$F(\overline{x} + h) = F'(\overline{x}) \left[h + \sum_{q=2}^{p-1} C_q h^q \right] + \mathcal{O}(h^p), \quad (7)$$

where $C_q = (1/q!)[F'(\overline{x})]^{-1} F^{(q)}(\overline{x})$, $q \geq 2$. We observe that $C_q h^q \in \mathbb{R}^n$ since $F^{(q)}(\overline{x}) \in \mathscr{L}(\mathbb{R}^n \times \cdots \times \mathbb{R}^n, \mathbb{R}^n)$ and $[F'(\overline{x})]^{-1} \in \mathscr{L}(\mathbb{R}^n)$.

In addition, we can express F' as

$$F'(\overline{x} + h) = F'(\overline{x}) \left[I + \sum_{q=2}^{p-1} q C_q h^{q-1} \right] + \mathcal{O}(h^{p-1}), \quad (8)$$

where I is the identity matrix. Therefore, $q C_q h^{q-1} \in \mathscr{L}(\mathbb{R}^n)$. From (8), we obtain

$$
\begin{aligned}
&\left[F'(\overline{x} + h) \right]^{-1} \\
&= \left[I + X_2 h + X_3 h^2 + X_4 h^4 + \cdots \right] \left[F'(\overline{x}) \right]^{-1} \\
&\quad + \mathcal{O}(h^{p-1}),
\end{aligned}
\quad (9)
$$

where

$$
\begin{aligned}
X_2 &= -2C_2, \\
X_3 &= 4C_2^2 - 3C_3, \\
X_4 &= -8C_2^3 + 6C_2 C_3 + 6C_3 C_2 - 4C_4, \\
&\quad \vdots
\end{aligned}
\quad (10)
$$

The equation

$$e^{(k+1)} = L e^{(k)^p} + \mathcal{O}\left(e^{(k)^{p+1}}\right), \quad (11)$$

where L is a p-linear function $L \in \mathscr{L}(\mathbb{R}^n \times \cdots \times \mathbb{R}^n, \mathbb{R}^n)$, is called the *error equation* and p is the *order of convergence*. Observe that $e^{(k)^p}$ is $(e^{(k)}, e^{(k)}, \ldots, e^{(k)})$.

We summarize the contents of this paper. In Section 2 we describe new iterative methods for solving nonlinear systems and show a procedure for constructing schemes with arbitrary order of convergence. The efficiency index and the computational efficiency index of our methods and of other known ones are analyzed in Section 3. Section 4 is devoted to numerical tests for confirming the theoretical results and to show the performance of the presented schemes. The examples used are a nonlinear one-dimensional heat conduction equation by transforming it in a big-sized nonlinear system by using finite differences and standard nonlinear systems. The paper finishes with some conclusions and the references used.

2. Development and Convergence of the Method

By adding a new step to Newton's method, we construct the following two-step scheme with fifth-order of convergence:

$$
\begin{aligned}
y^{(k)} &= x^{(k)} - \left[F'\left(x^{(k)}\right) \right]^{-1} F\left(x^{(k)}\right), \\
x^{(k+1)} &= y^{(k)} - \Bigg[\alpha_1 I + \alpha_2 \left[F'\left(y^{(k)}\right) \right]^{-1} F'\left(x^{(k)}\right) \\
&\quad + \alpha_3 \left(\left[F'\left(y^{(k)}\right) \right]^{-1} F'\left(x^{(k)}\right) \right)^2 \Bigg] \left[F'\left(y^{(k)}\right) \right]^{-1} \\
&\quad \cdot F\left(y^{(k)}\right),
\end{aligned}
\quad (12)
$$

where α_1, α_2, and α_3 are arbitrary parameters and I is the identity matrix of size $n \times n$.

The following result establishes the convergence of iterative method (12).

Theorem 1. *Let $F : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be sufficiently differentiable at each point of an open neighborhood D of $\overline{x} \in \mathbb{R}^n$, that is, a solution of the system $F(x) = 0$, and the initial estimation $x^{(0)}$ is close enough to $\overline{x}$. Let us suppose that $F'(x)$ is continuous and nonsingular in $\overline{x}$. Then, sequence $\{x^{(k)}\}_{k \geq 0}$ obtained from expression (12) converges to $\overline{x}$ with order 5 if the parameters take values $\alpha_1 = 5/4$, $\alpha_2 = -1/2$, $\alpha_3 = 1/4$, with the error equation in this case being*

$$e^{(k+1)} = \left[\frac{1}{2} C_2 C_3 C_2 - \frac{3}{2} C_3 C_2^2 \right] e^{(k)^5} + \mathcal{O}\left(e^{(k)^6}\right), \quad (13)$$

where $C_j = (1/j!)[F'(\overline{x})]^{-1} F^{(j)}(\overline{x})$, $j = 2, 3, \ldots$.

Proof. By using Taylor expansion of $F(x^{(k)})$ and $F'(x^{(k)})$ around $\overline{x}$,

$$
\begin{aligned}
F\left(x^{(k)}\right) &= F'(\overline{x}) \\
&\quad \cdot \left[e^{(k)} + C_2 e^{(k)^2} + C_3 e^{(k)^3} + C_4 e^{(k)^4} + C_5 e^{(k)^5} \right] \\
&\quad + \mathcal{O}\left(e^{(k)^6}\right), \\
F'\left(x^{(k)}\right) &= F'(\overline{x}) \\
&\quad \cdot \left[I + 2C_2 e^{(k)} + 3C_3 e^{(k)^2} + 4C_4 e^{(k)^3} + 5C_5 e^{(k)^4} \right] \\
&\quad + \mathcal{O}\left(e^{(k)^5}\right).
\end{aligned}
\quad (14)
$$

From the above expression, we have

$$
\begin{aligned}
&\left[F'\left(x^{(k)}\right) \right]^{-1} \\
&= \left[I + X_2 e^{(k)} + X_3 e^{(k)^2} + X_4 e^{(k)^3} + X_5 e^{(k)^4} \right] \\
&\quad \cdot \left[F'(\overline{x}) \right]^{-1} + \mathcal{O}\left(e^{(k)^5}\right),
\end{aligned}
\quad (15)
$$

where

$$X_2 = -2C_2,$$

$$X_3 = 4C_2^2 - 3C_3,$$

$$X_4 = -4C_4 + 6C_2C_3 + 6C_3C_2 - 8C_2^3, \qquad (16)$$

$$X_5 = -5C_5 + 8C_2C_4 - 12C_2^2C_3 + 9C_3^2 + 8C_4C_2$$
$$\quad - 12C_2C_3C_2 + 16C_2^4 - 12C_3C_2^2.$$

Then,

$$\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right) = e^{(k)} - C_2 e^{(k)^2} + 2\left(C_2^2 - C_3\right)$$
$$\cdot e^{(k)^3} + \left(4C_2C_3 + 3C_3C_2 - 4C_2^3 - 3C_4\right) e^{(k)^4}$$
$$+ \left(-4C_5 + 6C_2C_4 - 8C_2^2C_3 + 6C_3^2 + 4C_4C_2 \qquad (17)\right.$$
$$\left. - 6C_2C_3C_2 + 8C_2^4 - 6C_3C_2^2\right) e^{(k)^5} + \mathcal{O}\left(e^{(k)^6}\right).$$

So,

$$y^{(k)} - \overline{x} = C_2 e^{(k)^2} - 2\left(C_2^2 - C_3\right) e^{(k)^3} - \left(4C_2C_3\right.$$
$$\left. + 3C_3C_2 - 4C_2^3 - 3C_4\right) e^{(k)^4} - \left(-4C_5 + 6C_2C_4\right.$$
$$- 8C_2^2C_3 + 6C_3^2 + 4C_4C_2 - 6C_2C_3C_2 + 8C_2^4$$
$$\left. - 6C_3C_2^2\right) e^{(k)^5} + \mathcal{O}\left(e^{(k)^6}\right), \qquad (18)$$

$$\left(y^{(k)} - \overline{x}\right)^2 = C_2^2 e^{(k)^4} + 2C_2\left(C_3 - C_2^2\right) e^{(k)^5} + 2\left(C_3\right.$$
$$\left. - 2C_2^2\right) C_2 e^{(k)^5} + \mathcal{O}\left(e^{(k)^6}\right),$$

$$\left(y^{(k)} - \overline{x}\right)^3 = \mathcal{O}\left(e^{(k)^6}\right).$$

On the other hand,

$$F\left(y^{(k)}\right) = F'(\overline{x}) \left[\left(y^{(k)} - \overline{x}\right) + C_2\left(y^{(k)} - \overline{x}\right)^2\right]$$
$$+ \mathcal{O}\left(\left(y^{(k)} - \overline{x}\right)^3\right) = F'(\overline{x}) \left[C_2 e^{(k)^2} + 2\left(C_3 - C_2^2\right)\right.$$
$$\cdot e^{(k)^3} + \left(3C_4 + 5C_2^3 - 3C_3C_2 - 4C_2C_3\right) e^{(k)^4}$$
$$+ \left(4C_5 - 6C_2C_4 + 10C_2^2C_3 - 6C_3^2 - 4C_4C_2\right.$$
$$\left. + 8C_2C_3C_2 - 12C_2^4 + 6C_3C_2^2\right) e^{(k)^5}\right] + \mathcal{O}\left(e^{(k)^6}\right), \qquad (19)$$

$$F'\left(y^{(k)}\right) = F'(\overline{x}) \left[I + 2C_2^2 e^{(k)^2} + 4\left(C_2C_3 - 4C_2^3\right) e^{(k)^3}\right.$$
$$+ \left(6C_2C_4 + 8C_2^4 - 6C_2C_3C_2 - 8C_2^2C_3 + 3C_3C_2^2\right)$$
$$\left. \cdot e^{(k)^4}\right] + \mathcal{O}\left(e^{(k)^5}\right).$$

In a similar way as before, we obtain

$$\left[F'\left(y^{(k)}\right)\right]^{-1}$$
$$= \left[I + Y_2 e^{(k)} + Y_3 e^{(k)^2} + Y_4 e^{(k)^3} + Y_5 e^{(k)^4}\right] F'(\overline{x})^{-1} \qquad (20)$$
$$+ \mathcal{O}\left(e^{(k)^5}\right),$$

where

$$Y_2 = 0,$$

$$Y_3 = -2C_2^2,$$

$$Y_4 = 4C_2^3 - 4C_2C_3, \qquad (21)$$

$$Y_5 = -6C_2C_4 - 4C_2^4 + 6C_2C_3C_2 + 8C_2^2C_3 - 3C_3C_2^2,$$

so

$$\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right) = I + 2C_2 e^{(k)} + \left(3C_3 - 2C_2^2\right)$$
$$\cdot e^{(k)^2} + 4\left(C_4 - C_2C_3\right) e^{(k)^3} + \left(5C_5 + 2C_2^2C_3\right.$$
$$- 2C_2C_3C_2 - 3C_3C_2^2 + 4C_2^4 - 6C_2C_4\right) e^{(k)^4}$$
$$+ \mathcal{O}\left(e^{(k)^5}\right),$$

$$\left[\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right]^2 = I + 4C_2 e^{(k)} + 6C_3 e^{(k)^2}$$
$$+ \left(8C_4 - 2C_2C_3 + 6C_3C_2 - 8C_2^3\right) e^{(k)^3} + \left(10C_5\right.$$
$$- 10C_2^2C_3 - 12C_2C_3C_2 - 12C_3C_2^2 + 12C_2^4 \qquad (22)$$
$$\left. - 4C_2C_4 + 9C_3^2 + 8C_4C_2\right) e^{(k)^4} + \mathcal{O}\left(e^{(k)^5}\right),$$

$$\left[F'\left(y^{(k)}\right)\right]^{-1} F\left(y^{(k)}\right) = C_2 e^{(k)^2} + 2\left(C_3 - C_2^2\right) e^{(k)^3}$$
$$+ \left(3C_4 + 3C_2^3 - 3C_3C_2 - 4C_2C_3\right) e^{(k)^4} + \left(4C_5\right.$$
$$- 6C_2C_4 - 4C_4C_2 + 6C_2^2C_3 + 4C_2C_3C_2 + 6C_3C_2^2$$
$$\left. - 6C_3^2 - 4C_2^4\right) e^{(k)^5} + \mathcal{O}\left(e^{(k)^6}\right).$$

Therefore, we obtain

$$x^{(k+1)} = y^{(k)} - \left[\alpha_1 I + \alpha_2 \left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$
$$+ \alpha_3 \left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2 \left] \left[F'\left(y^{(k)}\right)\right]^{-1}\right.$$
$$\cdot F\left(y^{(k)}\right) = \overline{x} + \left(1 - \alpha_1 - \alpha_2 - \alpha_3\right) C_2 e^{(k)^2}$$
$$+ \left[2\left(-1 + \alpha_1 - \alpha_3\right) C_2^2 + 2\left(1 - \alpha_1 - \alpha_2 - \alpha_3\right) C_3\right]$$
$$\cdot e^{(k)^3} + \left[4\left(-1 + \alpha_1 - \alpha_3\right) C_2C_3\right.$$
$$+ 3\left(-1 + \alpha_1 - \alpha_3\right) C_3C_2$$

$$+ (4 - 3\alpha_1 + 3\alpha_2 + 5\alpha_3) C_2^3$$

$$+ 3 (1 - \alpha_1 - \alpha_2 - \alpha_3) C_4] e^{(k)4}$$

$$+ [4 (1 - \alpha_1 - \alpha_2 - \alpha_3) C_5$$

$$+ 6 (-1 + \alpha_1 - \alpha_3) C_2 C_4 + 4 (-1 + \alpha_1 - \alpha_3) C_4 C_2$$

$$+ (8 - 6\alpha_1 + 6\alpha_2 + 10\alpha_3) C_2^2 C_3$$

$$+ (6 - 4\alpha_1 + 6\alpha_2 + 10\alpha_3) C_2 C_3 C_2$$

$$+ 6 (1 - \alpha_1) C_3 C_2^2 + 6 (-1 + \alpha_1 - \alpha_3) C_3^2$$

$$+ (-8 + 4\alpha_1 - 6\alpha_2) C_2^4] e^{(k)5} + \mathcal{O}\left(e^{(k)6}\right).$$

$$(23)$$

If we replace, in the previous expression, the values of parameters $\alpha_1 = 5/4$, $\alpha_2 = -1/2$, and $\alpha_3 = 1/4$, we obtain the error equation:

$$e^{(k+1)} = \left[\frac{1}{2} C_2 C_3 C_2 - \frac{3}{2} C_3 C_2^2\right] e^{(k)5} + \mathcal{O}\left(e^{(k)6}\right), \quad (24)$$

and the proof is finished. □

If we add in expression (12) a third step with the same structure as the second one, we have

$$y^{(k)} = x^{(k)} - \left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right),$$

$$z^{(k)} = y^{(k)} - \left[\frac{5}{4}I - \frac{1}{2}\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$+ \frac{1}{4}\left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\left.\right]\left[F'\left(y^{(k)}\right)\right]^{-1}$$

$$\cdot F\left(y^{(k)}\right), \quad (25)$$

$$x^{(k+1)} = z^{(k)} - \left[\beta_1 I + \beta_2 \left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$+ \beta_3 \left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\left.\right]\left[F'\left(y^{(k)}\right)\right]^{-1}$$

$$\cdot F\left(z^{(k)}\right).$$

It can be proved that method (25) has eighth-order of convergence if the parameters take the values $\beta_1 = 3/2$, $\beta_2 = -1$, and $\beta_3 = 1/2$ and, for these values of parameters, the method is denoted as CCGT1. This idea can be generalized for obtaining an iterative method of arbitrary order of convergence. With each new step the order of convergence increases in three units, needing only one new functional evaluation.

Theorem 2. *Let* $t^{(k)} = \phi(x^{(k)})$ *be the iterative expression of a method of order* $p \geq 5$, *with asymptotic error constant M,*

where the two first steps are those of (12). By adding a new step in the form

$$x^{(k+1)} = t^{(k)} - \left[\beta_1 I + \beta_2 \left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$+ \beta_3 \left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\left.\right]\left[F'\left(y^{(k)}\right)\right]^{-1} \quad (26)$$

$$\cdot F\left(t^{(k)}\right),$$

then the order of the resulting method is $p+3$, *if* $\beta_1 = 3/2$, $\beta_2 = -1$, $\beta_3 = 1/2$, *with its error equation being*

$$e^{(k+1)} = (C_2 C_3 - 3 C_3 C_2) M e^{(k)p+3} + \mathcal{O}\left(e^{(k)p+4}\right), \quad (27)$$

where $C_j = (1/j!)[F'(\overline{x})]^{-1} F^{(j)}(\overline{x}), \; j = 2, 3, \ldots$.

Proof. Let us suppose that

$$t^{(k)} - \overline{x} = M e^{(k)p} + \mathcal{O}\left(e^{(k)p+1}\right), \quad (28)$$

where $M \neq 0$, $p \geq 5$.

Then,

$$F\left(t^{(k)}\right)$$

$$= F'(\overline{x})\left[t^{(k)} - \overline{x} + C_2 \left(t^{(k)} - \overline{x}\right)^2\right]$$

$$+ \mathcal{O}\left(\left(t^{(k)} - \overline{x}\right)^3\right)$$

$$= F'(\overline{x}) M e^{(k)p} + \mathcal{O}\left(e^{(k)p+1}\right), \quad (29)$$

$$\left[F'\left(y^{(k)}\right)\right]^{-1} F\left(t^{(k)}\right)$$

$$= M e^{(k)p} - 2 C_2^2 M e^{(k)p+2} + 4 C_2^3 M e^{(k)p+3}$$

$$- 4 C_2 C_3 M e^{(k)p+3} + \mathcal{O}\left(e^{(k)p+4}\right).$$

Therefore,

$$x^{(k+1)} = \overline{x} + [(1 - \beta_1 - \beta_2 - \beta_3) M] e^{(k)p}$$

$$+ [-2 (\beta_2 + 2\beta_3) C_2 M] e^{(k)p+1}$$

$$+ \left[2 (\beta_1 + 2\beta_2 + \beta_3) C_2^2 M - 3 (\beta_2 + 2\beta_3) C_3 M\right]$$

$$\cdot e^{(k)p+2} + \left[-4 (\beta_1 - 3\beta_3) C_2^3 M \right. \quad (30)$$

$$+ 2 (2\beta_1 + 4\beta_2 + 3\beta_3) C_2 C_3 M$$

$$\left. - 4 (\beta_2 + 2\beta_3) C_4 M - 6\beta_3 C_3 C_2 M\right] e^{(k)p+3}$$

$$+ \mathcal{O}\left(e^{(k)p+4}\right)$$

and, if we take $\beta_1 = 3/2$, $\beta_2 = -1$, $\beta_3 = 1/2$, we have order $p + 3$. In that case, the error equation is

$$e^{(k+1)} = (C_2 C_3 - 3C_3 C_2) M e^{(k)^{p+3}} + \mathcal{O}\left(e^{(k)^{p+4}}\right) \qquad (31)$$

and the proof is finished.   □

A small variation in the iterative expression of (12) allows us to construct another new scheme:

$$y^{(k)} = x^{(k)} - \left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right),$$

$$x^{(k+1)} = y^{(k)} - \left[\alpha_1 I + \alpha_2 \left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$\left. + \alpha_3 \left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\right] \left[F'\left(x^{(k)}\right)\right]^{-1} \qquad (32)$$

$$\cdot F\left(y^{(k)}\right),$$

whose order of convergence is described in the following result.

Theorem 3. *Let $F : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be sufficiently Fréchet differentiable at each point of an open neighborhood D of $\overline{x} \in \mathbb{R}^n$, that is, a solution of the system $F(x) = 0$, and the initial estimation $x^{(0)}$ is close enough to $\overline{x}$. Let us suppose that $F'(x)$ is continuous and nonsingular in $\overline{x}$. Then, sequence $\{x^{(k)}\}_{k \geq 0}$ obtained from expression (32) converges to $\overline{x}$ with order 5 if the parameters take the values $\alpha_1 = 1/4$, $\alpha_2 = 1/2$, and $\alpha_3 = 1/4$. In this case, the error equation is*

$$e^{(k+1)} = \frac{1}{2}\left(4C_2^4 - 3C_3 C_2^2 + C_2 C_3 C_2\right) e^{(k)^5}$$

$$+ \mathcal{O}\left(e^{(k)^6}\right). \qquad (33)$$

In a similar way as before, this structure can be extended in order to construct an iterative scheme of arbitrary order.

Theorem 4. *Let $t^{(k)} = \phi(x^{(k)})$ be the iterative expression of a method of order $p \geq 5$, with asymptotic error constant M, where the two first steps are those of (32). By adding a new step in the form*

$$x^{(k+1)} = t^{(k)} - \left[\beta_1 I + \beta_2 \left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$\left. + \beta_3 \left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\right] \left[F'\left(x^{(k)}\right)\right]^{-1} \qquad (34)$$

$$\cdot F\left(t^{(k)}\right),$$

then the order of the resulting method is $p+3$, if $\beta_1 = 1/2$, $\beta_2 = 0$, $\beta_3 = 1/2$, with the error equation being

$$e^{(k+1)} = \left(4C_2^3 + C_2 C_3 - 3C_3 C_2\right) M e^{(k)^{p+3}}$$

$$+ \mathcal{O}\left(e^{(k)^{p+4}}\right). \qquad (35)$$

TABLE 1: Efficiency index for different schemes.

Method	Order	Functional evaluations	I
CCGT1	8	$2n^2 + 3n$	$8^{1/(2n^2 + 3n)}$
CCGT2	8	$2n^2 + 3n$	$8^{1/(2n^2 + 3n)}$
SLB	6	$2n^2 + 2n$	$6^{1/(2n^2 + 2n)}$
XY	5	$2n^2 + 3n$	$5^{1/(2n^2 + 3n)}$
HMT	6	$2n^2 + 2n$	$6^{1/(2n^2 + 2n)}$

We will denote by CCGT2 the following three-step eighth-order iterative method:

$$y^{(k)} = x^{(k)} - \left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right),$$

$$z^{(k)} = y^{(k)} - \left[\frac{1}{4}I + \frac{1}{2}\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right.$$

$$\left. + \frac{1}{4}\left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\right]\left[F'\left(x^{(k)}\right)\right]^{-1} \qquad (36)$$

$$\cdot F\left(y^{(k)}\right),$$

$$x^{(k+1)} = z^{(k)} - \left[\frac{1}{2}I + \frac{1}{2}\left(\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right)\right)^2\right]$$

$$\cdot \left[F'\left(x^{(k)}\right)\right]^{-1} F\left(z^{(k)}\right).$$

3. Computational Efficiency

We are going to use two indices for comparing the different iterative schemes for solving nonlinear systems: the multidimensional extension of the efficiency index defined by Ostrowski as $I = p^{1/d}$ and the computational efficiency index CI defined in the Introduction as CI $= p^{1/(d+\text{op})}$, where p is the order of convergence, d is the number of functional evaluations per iteration, and op is the number of products-quotients per iteration. To compute F in any iterative method we need to calculate n scalar functions. The number of scalar functional evaluations is n^2 for any new evaluation of derivative F'.

In Table 1, the efficiency indices I of methods CCGT1, CCGT2, SLB, XY, and HMT are presented. The number of Jacobian evaluations is the same in all these schemes, but the number of functional evaluations and the order of convergence are different.

In Figure 1 we show the classical index defined by Ostrowski for the mentioned methods and systems of sizes from 2 to 50. We can observe that the index of CCGT1 and CCGT2 is the same as well as the index of SLB and HMT. For every size, except 2×2, the best index corresponds to method CCGT1.

On the other hand, in order to compute an inverse linear operator we solve a $n \times n$ linear system where we have to do $(1/3)n^3 + n^2 - (1/3)n$ products-quotients for obtaining LU decomposition and solving two triangular linear systems. In addition, we need n^2 products for matrix-vector multiplication.

TABLE 2: Functional evaluations and products-quotients of the methods.

Method	Order	NFE	NLS1	NLS2	$M \times V$	CI
CCGT1	8	$2n^2 + 3n$	1	6	4	$8^{1/((2/3)n^3+13n^2+(7/3)n)}$
CCGT2	8	$2n^2 + 3n$	3	4	2	$8^{1/((2/3)n^3+11n^2+(7/3)n)}$
SLB	6	$2n^2 + 2n$	2	3	3	$6^{1/((2/3)n^3+10n^2+(4/3)n)}$
XY	5	$2n^2 + 3n$	3	1	0	$5^{1/((2/3)n^3+6n^2+(7/3)n)}$
HMT	6	$2n^2 + 2n$	3	3	4	$6^{1/((2/3)n^3+12n^2+(4/3)n)}$

(a) Sizes from 2 to 20

(b) Sizes from 21 to 50
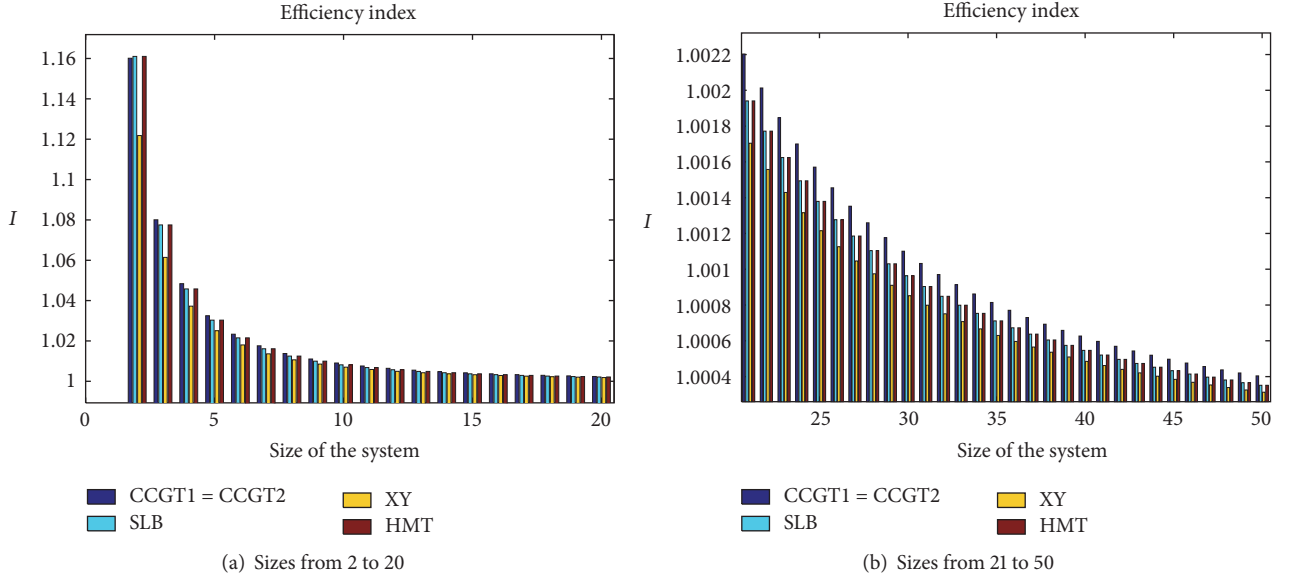
FIGURE 1: Efficiency index I for different sizes of the system.

Taking into account the previous considerations, we calculate CI of method CCGT1. For each iteration, we need to evaluate function F three times and twice Jacobian F', so $2n^2 + 3n$ functional evaluations are needed. In addition, we must solve one linear system with $F'(x^{(k)})$ as coefficients matrix (i.e., $(1/3)n^3 + n^2 - (1/3)n$ products-quotients), six linear systems with $F'(y^{(k)})$ as coefficients matrix (i.e., $(1/3)n^3 + 6n^2 - (1/3)n$ products-quotients), and four matrix-vector products ($4n^2$ products-quotients). Therefore, the value of index CI for method CCGT1 on a nonlinear system of size $n \times n$ is

$$\text{CI}_{\text{CCGT1}} = 8^{1/((2/3)n^3+13n^2+(7/3)n)}. \tag{37}$$

In Table 2, we show index CI of schemes CCGT1, CCGT2, SLB, XY, and HMT. In it, NFE is the number of functional evaluations, NLS1 denotes the number of linear systems with the matrix of coefficients $F'(x^{(k)})$ to be solved, NLS2 is the number of linear systems with another matrix of coefficients that are solved, and $M \times V$ denotes the number of products matrix-vector.

Let us observe that, although the classical index is similar in all these cases, it is not the case of the computational efficiency index since the number of inverse linear operators is different for each scheme. In Figure 2 the computational efficiency index for several methods and systems of sizes from 2 to 50 is shown. We can observe that until $n = 16$ the best index corresponds to method XY; meanwhile, for $n \geq 17$ the best index is the one corresponding to CCGT2.

4. Numerical Results

We check the numerical behavior of our method on a nonlinear one-dimensional heat conduction equation. A heat transfer problem is said to be one-dimensional if the temperature in the medium varies in one direction only and thus heat is transferred in one direction, and the variation of temperature and thus heat transfer in other directions are negligible or zero. For example, heat transfer through the glass of a window can be considered to be one-dimensional since heat transfer through the glass occurs predominantly in one direction (the direction normal to the surface of the glass) and heat transfer in other directions (from one side edge to the other and from the top edge to the bottom) is negligible.

To describe a heat transfer problem completely, an initial condition ($t = 0$) and two boundary conditions must be given for each direction of the coordinate system along which heat transfer is significant. Therefore, we need to specify two boundary conditions for one-dimensional problems, four boundary conditions for two-dimensional problems, and six boundary conditions for three-dimensional problems.

Different authors have approximated the solution of these problems by means of numerical techniques; see, for example, [11, 12] and the references therein.

In our study a particular case is used, corresponding to the following heat conduction equation:

$$u_{xx} = u_t + u_x - u^2 + f(x,t), \quad 0 \le x \le 1, \ t \ge 0, \quad (38)$$

where $f(x,t) = e^{-t}(-\pi \cos(\pi x) - (\pi^2 - 2)\sin(\pi x))$. The initial condition is $u(x,0) = \sin(\pi x)$ and the boundary conditions are

$$u(0,t) = 0,$$
$$u(1,t) = 0. \quad (39)$$

By applying an implicit method of finite differences we can transform problem (38) in a family of nonlinear systems, which provides the approximated solution in a time t_k from the approximated solution in t_{k-1}. We choose the spacial step $h = 1/nx$ and the temporal step $k = T_{\max}/nt$, where nx and nt are the number of x-subintervals and t-subintervals, respectively, and $T_{\max}$ is the final instant of our study, so we have selected a grid of domain $[0,1] \times [0, T_{\max}]$ with points (x_i, t_j),

$$x_i = 0 + ih, \quad i = 0, 1, \ldots, nx,$$
$$t_j = 0 + jk, \quad j = 0, 1, \ldots, nt. \quad (40)$$

We want to estimate the solution of (38) at these point, by transforming it in many nonlinear systems, as much as the number of t_j. To do that, we use the following approximations:

$$u_x(x,t) \approx \frac{u(x+h,t) - u(x-h,t)}{2h},$$
$$u_t(x,t) \approx \frac{u(x,t) - u(x,t-k)}{k}, \quad (41)$$
$$u_{xx}(x,t) \approx \frac{u(x+h,t) - 2u(x,t) + u(x-h,t)}{h^2}.$$

Denoting by $u_{i,j}$ the estimation of the solution at (x_i, t_j) and by replacing them in (38), we construct the following nonlinear system:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = \frac{u_{i,j} - u_{i,j-1}}{k} + \frac{u_{i+1,j} - u_{i-1,j}}{2h} \\ - u_{i,j}^2 + f(x_i, t_j), \quad (42)$$

for $i = 1, 2, \ldots, nx - 1$ and $t = 1, 2, \ldots, nt$. Some algebraic manipulations allow us to transform this system in

$$(2k - kh)u_{i+1,j} + (-4k - 2h^2)u_{i,j} + (2k + kh)u_{i-1,j} \\ + 2kh^2 u_{i,j}^2 = 2kh^2 f(x_i, t_j) - 2h^2 u_{i,j-1}, \quad (43)$$

TABLE 3: Numerical results for CCGT1 and different values of $T_{\max}$.

$T_{\max}$	nt	iter	Exact error	CPU time
0.1	10	2	0.0108	0.0921
0.1	100	1	0.0111	0.3576
0.3	10	3.6	0.0285	0.1411
0.3	100	1	0.0254	0.4389
0.5	10	4.4	0.0285	0.2042
0.5	100	1	0.0306	0.3533
0.7	10	4.7	0.0288	0.2103
0.7	100	1.29	0.0311	0.5365
1	10	5	0.0257	0.1830
1	500	1.58	0.0280	0.6246

for $i = 1, 2, \ldots, nx - 1$ and $j = 1, 2, \ldots, nt$. For a fixed j, we have the following nonlinear system of size $(nx - 1) \times (nx - 1)$:

$$(2k - kh)u_{2,j} - (4k + 2h^2)u_{1,j} \\ + 2kh^2 u_{1,j}^2 = 2kh^2 f(x_1, t_j) - 2h^2 u_{1,j-1},$$
$$(2k - kh)u_{i+1,j} - (4k + 2h^2)u_{i,j} + (2k + kh)u_{i-1,j} \\ + 2kh^2 u_{i,j}^2 = 2kh^2 f(x_i, t_j) - 2h^2 u_{i,j-1}, \quad (44)$$
$$i = 2, 3, \ldots, nx - 2,$$
$$- (4k + 2h^2)u_{nx-1,j} + (2k + kh)u_{nx-2,j} \\ + 2kh^2 u_{nx-1,j}^2 = 2kh^2 f(x_{nx-1}, t_j) - 2h^2 u_{nx-1,j-1}.$$

The unknowns of this system are $u_{1,j}, u_{2,j}, \ldots, u_{nx-1,j}$, that is, the approximations of the solution in t_j. We observe that for solving this system we need the solution in t_{j-1}.

We are going to solve this system for different values of $T_{\max}$, using in each case $nt = 10$ or higher, $nx = 200$, and methods CCGT1 and CCGT2. As initial guess, we use the solution at t_{j-1}. We compare the obtained values with the exact solution $u(x,t) = e^{-t}\sin(\pi x)$, in order to analyze the stability and consistence of the new method.

All computations are performed in the programming package MATLAB $R2014b$ using variable precision arithmetic with 50 digits of mantissa. For every value of $T_{\max}$, we analyze the mean number of iterations (iter) needed to converge to the solution such that $\|F(x^{(k+1)})\| < 10^{-14}$ is satisfied, where $\| \cdot \|$ denotes the Euclidean norm. Let us recall that the iterative methods are applied for solving the nonlinear system that involves each column of the solution matrix; the mean of the number of iterations needed when all the columns have been calculated is shown in the tables. The processor of the machine used is Intel(R) Xeon(R) CPU E5-2420 v2 @ 2.20 GHz, with 64 GB of RAM. The results are presented in Tables 3 and 4, where the good performance of both methods in terms of exact error and CPU time (in seconds) is observed, even for $nt = 10$.
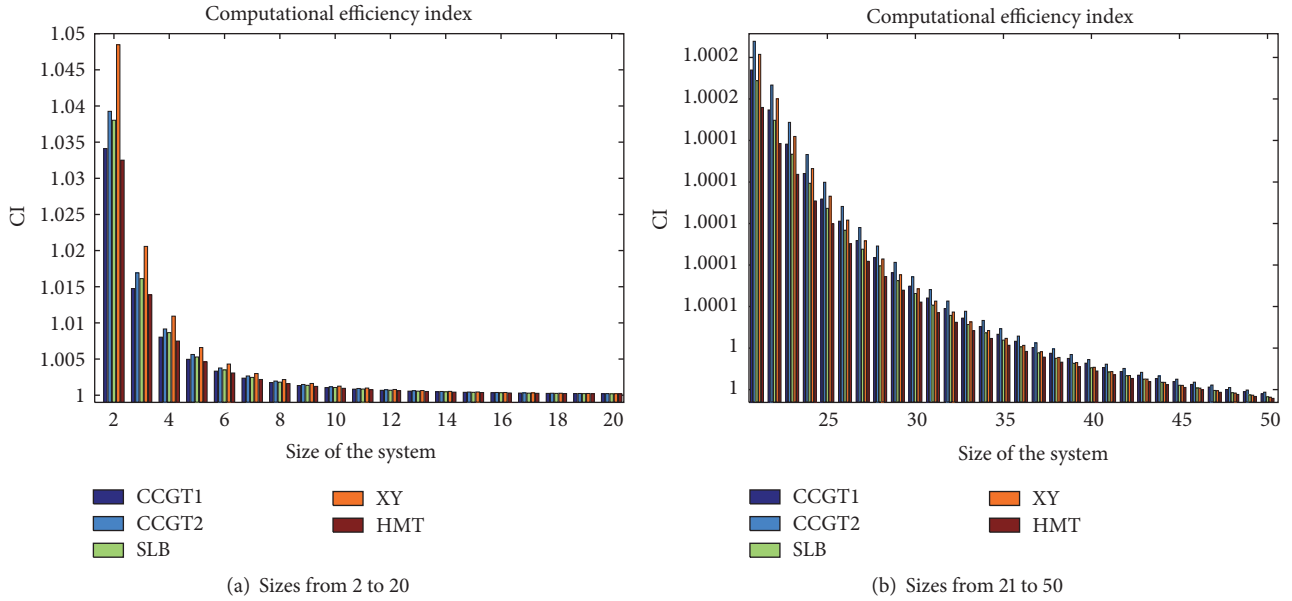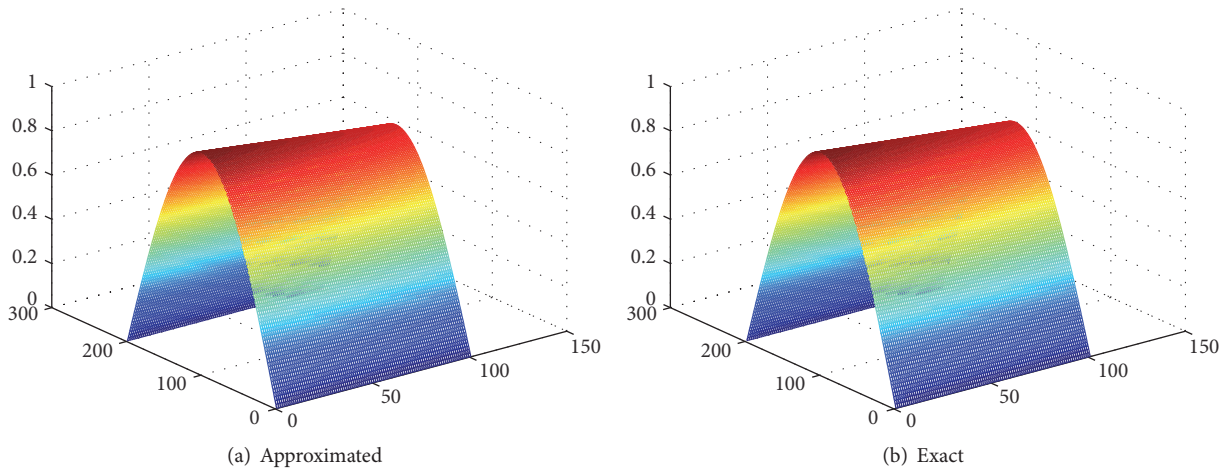
(a) Sizes from 2 to 20

(b) Sizes from 21 to 50

FIGURE 2: Index CI for different sizes of the system.

(a) Approximated

(b) Exact

FIGURE 3: Graphics of the approximated and exact solutions for $t \in [0, 0.1]$.

In Figure 3(a) we show the approximated solution of the problem when $T_{max} = 0.1$, by taking $nt = 100$ and $nx = 200$ and using method CCGT1. It is a good approximation if we compare it with the exact solution (Figure 3(b)). In Figure 4 the absolute value of the exact error is presented.

In the rest of the section, the new method CCGT1 is compared with methods SLB, HMT, and XY for solving some academical nonlinear systems. The numerical results are shown in Tables 5, 6, and 7. All experiments have been carried out on MATLAB $R2014b$ using variable precision arithmetic with 2000 digits of mantissa. To verify the theoretical order of convergence p, we calculate the approximated computational order of convergence (ACOC) introduced in [13] as

$$p \approx \text{ACOC} = \frac{\ln \left(\left\| x^{(k+1)} - x^{(k)} \right\| / \left\| x^{(k)} - x^{(k-1)} \right\| \right)}{\ln \left(\left\| x^{(k)} - x^{(k-1)} \right\| / \left\| x^{(k-1)} - x^{(k-2)} \right\| \right)}. \quad (45)$$

Example 1. The first nonlinear system is defined by (see [9])

$$x_i - \cos \left(2x_i - \sum_{j=1}^{n} x_j \right) = 0, \quad i = 1, 2, \dots, n. \quad (46)$$

In this test example we use $n = 4$ and the initial estimation $x^{(0)} = (0.75, 0.75, 0.75, 0.75)^T$, with the solution being $\overline{x} \approx (0.5149, 0.5149, 0.5149, 0.5149)^T$. In Table 5 we show the values of $\|x^{(k)} - x^{(k-1)}\|$ and $\|F(x^{(k)})\|$ for $k = 1, 2, 3$ as well as the value of ACOC.

Example 2. The second nonlinear system is also defined in [9]:

$$x_i^2 x_{i+1} - 1 = 0, \quad 1 \geq i \geq n - 1,$$

$$x_n^2 x_1 - 1 = 0. \quad (47)$$
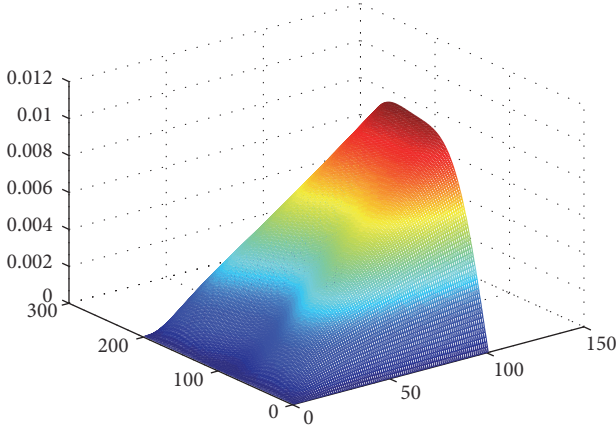
FIGURE 4: Graphic of the exact error for $t \in [0, 0.1]$.

TABLE 4: Numerical results for CCGT2 and different values of $T_{\max}$.

$T_{\max}$	nt	iter	Exact error	CPU time
0.1	10	2	0.0108	0.0955
0.1	100	1	0.0111	0.3578
0.3	10	3.6	0.0238	0.1675
0.3	100	1	0.0254	0.4075
0.5	10	4.3	0.0285	0.1951
0.5	100	1	0.0306	0.3808
0.7	10	4.7	0.0288	0.1790
0.7	100	1.29	0.0311	0.5092
1	10	5	0.0257	0.2198
1	100	1.58	0.0280	0.5280

The numerical results are displayed in Table 6. The initial estimation is $x^{(0)} = (1.5, 1.5, \ldots, 1.5)^T$ and the size of the system is $n = 199$, with the solution being $\overline{x} = (1, 1, \ldots, 1)^T$. We show the same information as in the previous example.

Example 3. Finally, the third example is

$$
\begin{aligned}
x_2 x_3 + x_4 (x_2 + x_3) &= 0, \\
x_1 x_3 + x_4 (x_1 + x_3) &= 0, \\
x_1 x_2 + x_4 (x_1 + x_2) &= 0, \\
x_1 x_2 + x_1 x_3 + x_2 x_3 - 1 &= 0.
\end{aligned}
\tag{48}
$$

The solution of this system is $\overline{x} \approx (0.577350, 0.577350, 0.577350, -0.284675)^T$. The numerical results of this example by using the initial estimation $x^{(0)} = (1, 1, 1, -1)^T$ and $x^{(0)} = (1, 1, 1, 0)^T$ are displayed in Tables 7 and 8, respectively. For the first initial guess, the results are very bad, and only method CCGT1 is convergent. For the second one, the numerical solutions are standard and confirm the theoretical results.

TABLE 5: Numerical results for Example 1.

	CCGT1	SLB	HMT	XY
$\|x^{(1)} - x^{(0)}\|$	0.4701	0.4701	0.4701	0.4672
$\|x^{(2)} - x^{(1)}\|$	$1.84e - 8$	$1.10e - 5$	$1.13e - 5$	$2.96e - 3$
$\|x^{(3)} - x^{(2)}\|$	$1.99e - 66$	$6.24e - 33$	$7.34e - 33$	$6.31e - 14$
ACOC	7.8255	5.8832	5.8837	4.8546
$\|F(x^{(1)})\|$	$5.00e - 8$	$2.98e - 5$	$3.06e - 5$	$8.04e - 3$
$\|F(x^{(2)})\|$	$5.40e - 66$	$1.69e - 32$	$1.99e - 32$	$1.71e - 13$
$\|F(x^{(3)})\|$	$1.01e - 529$	$5.67e - 196$	$1.52e - 195$	$7.61e - 67$

TABLE 6: Numerical results for Example 2.

	CCGT1	SLB	HMT	XY
$\|x^{(1)} - x^{(0)}\|$	7.05	7.03	7.01	6.75
$\|x^{(2)} - x^{(1)}\|$	$1.46e - 3$	$2.79e - 2$	$4.61e - 2$	$3.06e - 1$
$\|x^{(3)} - x^{(2)}\|$	$4.18e - 32$	$3.76e - 15$	$1.06e - 14$	$6.16e - 7$
ACOC	7.7499	5.7777	5.7935	4.2401
$\|F(x^{(1)})\|$	$4.39e - 3$	$8.39e - 2$	$1.39e - 1$	$9.38e - 1$
$\|F(x^{(2)})\|$	$1.25e - 31$	$1.13e - 15$	$3.18e - 14$	$1.85e - 6$
$\|F(x^{(3)})\|$	$5.54e - 260$	$6.71e - 99$	$5.07e - 90$	$7.18e - 35$

TABLE 7: Numerical results for Example 3 and initial guess $x^{(0)} = (1, 1, 1, -1)^T$.

	CCGT1	SLB	HMT	XY
$\|x^{(1)} - x^{(0)}\|$	1.04	—	—	—
$\|x^{(2)} - x^{(1)}\|$	$5.12e - 2$	—	—	—
$\|x^{(3)} - x^{(2)}\|$	$1.76e - 9$	—	—	—
ACOC	5.7069	—	—	—
$\|F(x^{(1)})\|$	$7.59e - 2$	—	—	—
$\|F(x^{(2)})\|$	$1.16e - 9$	—	—	—
$\|F(x^{(3)})\|$	$2.03e - 55$	—	—	—

TABLE 8: Numerical results for Example 3 and initial guess $x^{(0)} = (1, 1, 1, 0)^T$.

	CCGT1	SLB	HMT	XY
$\|x^{(1)} - x^{(0)}\|$	$7.87e - 1$	$7.85e - 1$	$7.84e - 1$	$7.64e - 1$
$\|x^{(2)} - x^{(1)}\|$	$1.52e - 4$	$2.82e - 3$	$3.99e - 3$	$3.00e - 2$
$\|x^{(3)} - x^{(2)}\|$	$2.03e - 45$	$4.54e - 19$	$4.98e - 20$	$8.55e - 9$
ACOC	11.008	6.4608	7.3703	4.6578
$\|F(x^{(1)})\|$	$3.24e - 4$	$6.18e - 3$	$8.73e - 3$	$6.86e - 2$
$\|F(x^{(2)})\|$	$4.09e - 45$	$9.23e - 19$	$1.01e - 19$	$1.76e - 8$
$\|F(x^{(3)})\|$	$5.31e - 463$	$1.81e - 117$	$3.43e - 143$	$6.66e - 44$

5. Conclusions

We have designed two three-step iterative methods of order eight for solving nonlinear systems with a competitive efficiency index. Moreover, higher order methods can be constructed by adding new steps with the same structure involving one new functional evaluation (per step) and increasing the order of convergence in three units per step. These methods are specially useful in nonlinear systems with a big size. Their performance has been checked by means

of a heat transfer problem showing excellent results and also has been compared with recent high-order methods on academical examples.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

 [1] F. Awawdeh, "On new iterative method for solving systems of nonlinear equations," *Numerical Algorithms*, vol. 54, no. 3, pp. 395–409, 2010.

 [2] A. Cordero, J. L. Hueso, E. Martínez, and J. R. Torregrosa, "A modified Newton-Jarratt's composition," *Numerical Algorithms*, vol. 55, no. 1, pp. 87–99, 2010.

 [3] C. Grosan and A. Abraham, "A new approach for solving nonlinear equations systems," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 38, no. 3, pp. 698–714, 2008.

 [4] J. L. Hueso, E. Martínez, and C. Teruel, "Convergence, efficiency and dynamics of new fourth and sixth order families of iterative methods for nonlinear systems," *Journal of Computational and Applied Mathematics*, vol. 275, pp. 412–420, 2015.

 [5] H. Montazeri, F. Soleymani, S. Shateyi, and S. Motsa, "On a new method for computing the numerical solution of systems of nonlinear equations," *Journal of Applied Mathematics*, vol. 2012, Article ID 751975, 15 pages, 2012.

 [6] J. R. Sharma and H. Arora, "Efficient Jarratt-like methods for solving systems of nonlinear equations," *Calcolo. A Quarterly on Numerical Analysis and Theory of Computation*, vol. 51, no. 1, pp. 193–210, 2014.

 [7] F. Soleymani, T. Lotfi, and P. Bakhtiari, "A multi-step class of iterative methods for nonlinear systems," *Optimization Letters*, vol. 8, no. 3, pp. 1001–1015, 2014.

 [8] X. Y. Xiao and H. W. Yin, "Increasing the order of convergence for iterative methods to solve nonlinear systems," *Calcolo. A Quarterly on Numerical Analysis and Theory of Computation*, vol. 53, no. 3, pp. 285–300, 2016.

 [9] X. Wang, T. Zhang, W. Qian, and M. Teng, "Seventh-order derivative-free iterative method for solving nonlinear systems," *Numerical Algorithms*, vol. 70, no. 3, pp. 545–558, 2015.

[10] A. M. Ostrowski, *Solution of Equations and Systems of Equations*, Prentice-Hall, Englewood Cliffs, NY, USA, 1964.

[11] M. A. Christou, C. Sophocleous, and C. I. Christov, "Numerical investigation of the nonlinear heat diffusion equation with high nonlinearity on the boundary," *Applied Mathematics and Computation*, vol. 201, no. 1-2, pp. 729–738, 2008.

[12] R. Kouhia, "On the solution of non-linear diffusion equation," *Rakenteiden Mekaniikka (Journal of Structural Mechanics)*, vol. 46, no. 4, pp. 116–130, 2013.

[13] A. Cordero and J. R. Torregrosa, "Variants of Newton's method using fifth-order quadrature formulas," *Applied Mathematics and Computation*, vol. 190, no. 1, pp. 686–698, 2007.