MDPI

Chaos and Stability in a New Iterative Family for Solving Nonlinear Equations

Alicia Cordero *[ID], Marlon Moscoso-Martínez [ID] and Juan R. Torregrosa [ID]

Institute for Multidisciplinary Mathematics, Universitat Politècnica de València, Camino de Vera s/n, 46022 València, Spain; marmosma@doctor.upv.es (M.M.-M.); jrtorre@mat.upv.es (J.R.T.)
* Correspondence: acordero@mat.upv.es

Abstract: In this paper, we present a new parametric family of three-step iterative for solving nonlinear equations. First, we design a fourth-order triparametric family that, by holding only one of its parameters, we get to accelerate its convergence and finally obtain a sixth-order uniparametric family. With this last family, we study its convergence, its complex dynamics (stability), and its numerical behavior. The parameter spaces and dynamical planes are presented showing the complexity of the family. From the parameter spaces, we have been able to determine different members of the family that have bad convergence properties, as attracting periodic orbits and attracting strange fixed points appear in their dynamical planes. Moreover, this same study has allowed us to detect family members with especially stable behavior and suitable for solving practical problems. Several numerical tests are performed to illustrate the efficiency and stability of the presented family.

Keywords: nonlinear equations; multistep iterative methods; convergence analysis; complex dynamics; chaos and stability

check for updates

1. Introduction

Many problems in Computational Sciences and other disciplines can be stated in the form of a nonlinear equation or nonlinear systems using mathematical modeling. In particular, a large number of problems in Applied Mathematics and Engineering are solved by finding the solutions of these equations.

In the literature, there are many methods and families of iterative schemes that have been designed by using different procedures to approximate the simple roots of a nonlinear equation $f(x) = 0$, where $f : I \subseteq \mathbb{R} \to \mathbb{R}$ is a real function defined in an open interval I. We can find in [1–3] several surveys and overviews of the iterative schemes published in the last years. Each method has a different behavior. This behavior is characterized with the efficiency criteria and the complex dynamics tools.

In this paper, we introduce a new family of multistep iterative schemes to solve nonlinear equations, which contains as an element of this family, a particular method presented in [4]. This family is built from the Ostrowski's scheme, adding a Newton step with a "frozen" derivative and using a divided difference operator. Therefore, the family has a three-step iterative expression. Furthermore, it has three arbitrary parameters named α, β, and γ, which can take real or complex values, and an order of convergence of at least four. The order of convergence will be discussed in Section 2.

From the error equation we observe by fixing two parameters in function of the third one, an uniparametric family of sixth-order iterative methods is obtained. We analyze the dynamical behavior of this family in terms of values of the parameter, in order to detect its elements with good stability properties and others with chaotic behavior. The concept of chaos has been widely discussed (see, for example, in [5]) and it is commonly understood as the presence of complex orbit structure and extreme sensitivity of orbits to small perturbations. Moreover, the presence of unstable periodic orbits of all periods

is also included in the concept of chaotic system. For this study, we use tools of discrete complex dynamics that we introduce in Section 3.

In Section 4, we present the performance of the presented schemes on several test functions. These numerical tests allow us to confirm the results obtained in the dynamical section and to compare our schemes with other known ones. The manuscript finishes with some conclusions and the references used in it.

The parametric family object of study in this manuscript has the following iterative expression:

$$
\begin{cases}
y_k = x_k - \dfrac{f(x_k)}{f'(x_k)}, \\[2mm]
z_k = y_k - \dfrac{f(y_k)}{2f[x_k, y_k] - f'(x_k)}, \\[2mm]
x_{k+1} = z_k - (\alpha + \beta u_k + \gamma v_k)\dfrac{f(z_k)}{f'(x_k)},
\end{cases}
\tag{1}
$$

where $u_k = 1 - \dfrac{f[x_k, y_k]}{f'(x_k)}$; $v_k = \dfrac{f'(x_k)}{f[x_k, y_k]}$; $k = 0, 1, 2, ...$; and α, β, and γ are arbitrary parameters.

The divided difference operator $f[\cdot, \cdot] : I \times I \subset \mathbb{R} \times \mathbb{R} \to \mathcal{L}(\mathbb{R})$ defined by Ortega and Rheinboldt in [6], satisfies

$$
f[x, y](x - y) = f(x) - f(y), \quad \forall x, y \in I.
\tag{2}
$$

2. Convergence of the New Family

In this section, we perform the convergence analysis of the new triparametric iterative family. Furthermore, we propose a strategy to reduce the triparametric scheme to an uniparametric scheme in order to accelerate the convergence.

Theorem 1. *Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be a sufficiently differentiable function on an open interval I and $\xi \in I$ a simple root of the nonlinear equation $f(x) = 0$. Suppose that $f(x)$ is continuous and sufficiently differentiable in an environment of the simple root ξ, and x_0 is an initial estimate close enough to ξ. Then, the sequence $\{x_k\}_{k \geq 0}$ obtained by using the expression (1) converges to ξ with an order of convergence of four, being its error equation*

$$
e_{k+1} = (1 - \alpha - \gamma)C_2\left(C_2^2 - C_3\right)e_k^4 + \mathcal{O}\left(e_k^5\right),
$$

where $e_k = x_k - \xi$, $C_q = \dfrac{1}{q!}\dfrac{f^{(q)}(\xi)}{f'(\xi)}$ and $q = 2, 3, ...$

Proof. Let ξ be a simple root of $f(x)$ (that is, $f(\xi) = 0$ and $f'(\xi) \neq 0$) and $x_k = \xi + e_k$. Using Taylor expansion of $f(x_k)$ and $f'(x_k)$ around ξ, we have

$$
\begin{aligned}
f(x_k) &= f(\xi + e_k) \\
&= f(\xi) + f'(\xi)e_k + \frac{1}{2!}f''(\xi)e_k^2 + \frac{1}{3!}f'''(\xi)e_k^3 + \frac{1}{4!}f^{(iv)}(\xi)e_k^4 + \mathcal{O}(e_k^5) \\
&= f'(\xi)\left[e_k + \frac{1}{2!}\frac{f''(\xi)}{f'(\xi)}e_k^2 + \frac{1}{3!}\frac{f'''(\xi)}{f'(\xi)}e_k^3 + \frac{1}{4!}\frac{f^{(iv)}(\xi)}{f'(\xi)}e_k^4 + \mathcal{O}(e_k^5)\right] \\
&= f'(\xi)\left[e_k + C_2 e_k^2 + C_3 e_k^3 + C_4 e_k^4 + \mathcal{O}(e_k^5)\right],
\end{aligned}
\tag{3}
$$

and

$$
\begin{aligned}
f'(x_k) &= f'(\xi + e_k) \\
&= f'(\xi) + f''(\xi)e_k + \frac{1}{2!}f'''(\xi)e_k^2 + \frac{1}{3!}f^{(iv)}(\xi)e_k^3 + \mathcal{O}(e_k^4) \\
&= f'(\xi)\left[1 + \frac{f''(\xi)}{f'(\xi)}e_k + \frac{1}{2!}\frac{f'''(\xi)}{f'(\xi)}e_k^2 + \frac{1}{3!}\frac{f^{(iv)}(\xi)}{f'(\xi)}e_k^3 + \mathcal{O}(e_k^4)\right] \\
&= f'(\xi)\left[1 + 2C_2 e_k + 3C_3 e_k^2 + 4C_4 e_k^3 + \mathcal{O}(e_k^4)\right],
\end{aligned}
\tag{4}
$$

where $C_q = \dfrac{1}{q!}\dfrac{f^{(q)}(\xi)}{f'(\xi)}$, $q = 2, 3, \dots$

Dividing (3) by (4), we get

$$
\frac{f(x_k)}{f'(x_k)} = e_k - C_2 e_k^2 + 2\left(C_2^2 - C_3\right)e_k^3 - \left(4C_2^3 - 7C_2 C_3 + 3C_4\right)e_k^4 + \mathcal{O}\left(e_k^5\right). \tag{5}
$$

Replacing (5) in the first step of family (1), we have

$$
y_k = \xi + C_2 e_k^2 - 2\left(C_2^2 - C_3\right)e_k^3 + \left(4C_2^3 - 7C_2 C_3 + 3C_4\right)e_k^4 + \mathcal{O}\left(e_k^5\right). \tag{6}
$$

Using Taylor expansion again, similar to (3), to develop $f(y_k)$ around ξ, we get

$$
f(y_k) = f'(\xi)\left[C_2 e_k^2 - 2\left(C_2^2 - C_3\right)e_k^3 + \left(5C_2^3 - 7C_2 C_3 + 3C_4\right)e_k^4 + \mathcal{O}\left(e_k^5\right)\right]. \tag{7}
$$

With (3), (6), and (7), we calculate the divided difference operator defined in (2), obtaining

$$
f[x_k, y_k] = f'(\xi)\left[1 + C_2 e_k + \left(C_2^2 + C_3\right)e_k^2 - \left(2C_2^3 - 3C_2 C_3 - C_4\right)e_k^3 + \mathcal{O}\left(e_k^4\right)\right]. \tag{8}
$$

Then, substituting (3), (4), (6), and (8) in the second step of family (1), we have

$$
z_k = \xi + \left(C_2^3 - C_2 C_3\right)e_k^4 + \mathcal{O}\left(e_k^5\right). \tag{9}
$$

Using Taylor series once again, similar to (3), to expand $f(z_k)$ around ξ, we get

$$
f(z_k) = f'(\xi)\left[\left(C_2^3 - C_2 C_3\right)e_k^4 + \mathcal{O}\left(e_k^5\right)\right]. \tag{10}
$$

Replacing (4) and (8) in u_k and v_k of family (1), we have

$$
u_k = C_2 e_k - \left(3C_2^2 - 2C_3\right)e_k^2 + \left(8C_2^3 - 10C_2 C_3 + 3C_4\right)e_k^3 + \mathcal{O}\left(e_k^4\right), \tag{11}
$$

$$
v_k = 1 + C_2 e_k - 2\left(C_2^2 - C_3\right)e_k^2 + 3\left(C_2^3 - 2C_2 C_3 + C_4\right)e_k^3 + \mathcal{O}\left(e_k^4\right). \tag{12}
$$

Finally, substituting (4), (9)–(12) in the third step of family (1), we get

$$
x_{k+1} = \xi + (1 - \alpha - \gamma)C_2\left(C_2^2 - C_3\right)e_k^4 + \mathcal{O}\left(e_k^5\right), \tag{13}
$$

being the error equation

$$
e_{k+1} = (1 - \alpha - \gamma)C_2\left(C_2^2 - C_3\right)e_k^4 + \mathcal{O}\left(e_k^5\right), \tag{14}
$$

and the proof is finished. □

From Theorem 1, it follows that the new triparametric family of iterative methods has an order of convergence of four for any real or complex values of the parameters α, β, and

γ. However, convergence can be speed-up if only one parameter is held and the family is reduced to an uniparametric iterative scheme. The latter can be seen in Theorem 2.

Theorem 2. *Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be a sufficiently differentiable function on an open interval I and $\xi \in I$ a simple root of the nonlinear equation $f(x) = 0$. Suppose that $f(x)$ is continuous and sufficiently differentiable in an environment of the simple root ξ, and x_0 is an initial estimate close enough to ξ. Then, the sequence $\{x_k\}_{k \geq 0}$ obtained by using the expression (1) converges to ξ with an order of convergence of six, provided that $\beta = 1 + \alpha$ and $\gamma = 1 - \alpha$, being its error equation*

$$e_{k+1} = \left(6C_2^5 - 7C_2^3 C_3 + C_2 C_3^2 \right) e_k^6 + \mathcal{O}\left(e_k^7\right),$$

where $e_k = x_k - \xi$, $C_q = \dfrac{1}{q!} \dfrac{f^{(q)}(\xi)}{f'(\xi)}$ and $q = 2, 3, \ldots$

Proof. Let ξ be a simple root of $f(x)$ (that is, $f(\xi) = 0$ and $f'(\xi) \neq 0$) and $x_k = \xi + e_k$. Using Taylor expansion of $f(x_k)$ and $f'(x_k)$ around ξ, we have

$$
\begin{aligned}
f(x_k) &= f(\xi + e_k) \\
&= f(\xi) + f'(\xi)e_k + \frac{1}{2!}f''(\xi)e_k^2 + \cdots + \frac{1}{6!}f^{(vi)}(\xi)e_k^6 + \mathcal{O}(e_k^7) \\
&= f'(\xi)\left[e_k + \frac{1}{2!}\frac{f''(\xi)}{f'(\xi)}e_k^2 + \cdots + \frac{1}{6!}\frac{f^{(vi)}(\xi)}{f'(\xi)}e_k^6 + \mathcal{O}(e_k^7) \right] \\
&= f'(\xi)\left[e_k + C_2 e_k^2 + C_3 e_k^3 + C_4 e_k^4 + C_5 e_k^5 + C_6 e_k^6 + \mathcal{O}(e_k^7) \right],
\end{aligned}
\tag{15}
$$

and

$$
\begin{aligned}
f'(x_k) &= f'(\xi + e_k) \\
&= f'(\xi) + f''(\xi)e_k + \frac{1}{2!}f'''(\xi)e_k^2 + \cdots + \frac{1}{5!}f^{(vi)}(\xi)e_k^5 + \mathcal{O}(e_k^6) \\
&= f'(\xi)\left[1 + \frac{f''(\xi)}{f'(\xi)}e_k + \frac{1}{2!}\frac{f'''(\xi)}{f'(\xi)}e_k^2 + \cdots + \frac{1}{5!}\frac{f^{(vi)}(\xi)}{f'(\xi)}e_k^5 + \mathcal{O}(e_k^6) \right] \\
&= f'(\xi)\left[1 + 2C_2 e_k + 3C_3 e_k^2 + 4C_4 e_k^3 + 5C_5 e_k^4 + 6C_6 e_k^5 + \mathcal{O}(e_k^6) \right],
\end{aligned}
\tag{16}
$$

where $C_q = \dfrac{1}{q!}\dfrac{f^{(q)}(\xi)}{f'(\xi)}$, $q = 2, 3, \ldots$

Dividing (15) by (16), we get

$$
\begin{aligned}
\frac{f(x_k)}{f'(x_k)} = {}& e_k - C_2 e_k^2 + 2\left(C_2^2 - C_3 \right)e_k^3 - \left(4C_2^3 - 7C_2 C_3 + 3C_4 \right)e_k^4 + \\
& \left(8C_2^4 - 20C_2^2 C_3 + 6C_3^2 + 10C_2 C_4 - 4C_5 \right)e_k^5 - \left(16C_2^5 - 52C_2^3 C_3 + \right. \\
& \left. 28C_2^2 C_4 - 17C_3 C_4 + C_2\left(33C_3^2 - 13C_5 \right) + 5C_6 \right)e_k^6 + \mathcal{O}\left(e_k^7\right).
\end{aligned}
\tag{17}
$$

Replacing (17) in the first step of family (1), we have

$$
\begin{aligned}
y_k = {}& \xi + C_2 e_k^2 - 2\left(C_2^2 - C_3 \right)e_k^3 + \left(4C_2^3 - 7C_2 C_3 + 3C_4 \right)e_k^4 - \\
& \left(8C_2^4 - 20C_2^2 C_3 + 6C_3^2 + 10C_2 C_4 - 4C_5 \right)e_k^5 + \left(16C_2^5 - 52C_2^3 C_3 + \right. \\
& \left. 28C_2^2 C_4 - 17C_3 C_4 + C_2\left(33C_3^2 - 13C_5 \right) + 5C_6 \right)e_k^6 + \mathcal{O}\left(e_k^7\right).
\end{aligned}
\tag{18}
$$

Using Taylor expansion again, similar to (15), to expand $f(y_k)$ around ξ, we get

$$
\begin{aligned}
f(y_k) = f'(\xi)\Big[& C_2 e_k^2 - 2\Big(C_2^2 - C_3\Big)e_k^3 + \Big(5C_2^3 - 7C_2 C_3 + 3C_4\Big)e_k^4 - \\
& 2\Big(6C_2^4 - 12C_2^2 C_3 + 3C_3^2 + 5C_2 C_4 - 2C_5\Big)e_k^5 + \Big(28C_2^5 - 73C_2^3 C_3 + \\
& 34C_2^2 C_4 - 17C_3 C_4 + C_2\Big(37C_3^2 - 13C_5\Big) + 5C_6\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big)\Big].
\end{aligned}
\tag{19}
$$

With (15), (18), and (19), we calculate the divided difference operator defined in (2), obtaining

$$
\begin{aligned}
f[x_k, y_k] = f'(\xi)\Big[& 1 + C_2 e_k + \Big(C_2^2 + C_3\Big)e_k^2 - \Big(2C_2^3 - 3C_2 C_3 - C_4\Big)e_k^3 + \\
& \Big(4C_2^4 - 8C_2^2 C_3 + 2C_3^2 + 4C_2 C_4 + C_5\Big)e_k^4 + \Big(-8C_2^5 + 20C_2^3 C_3 - \\
& 11C_2^2 C_4 + 5C_3 C_4 + C_2\Big(-9C_3^2 + 5C_5\Big) + C_6\Big)e_k^5 + \mathcal{O}\Big(e_k^6\Big)\Big].
\end{aligned}
\tag{20}
$$

Then, substituting (15), (16), (18), and (20) in the second step of family (1), we have

$$
\begin{aligned}
z_k = \xi + & \Big(C_2^3 - C_2 C_3\Big)e_k^4 - 2\Big(2C_2^4 - 4C_2^2 C_3 + C_3^2 + C_2 C_4\Big)e_k^5 + \\
& \Big(10C_2^5 - 30C_2^3 C_3 + 12C_2^2 C_4 - 7C_3 C_4 + 3C_2\Big(6C_3^2 - C_5\Big)\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big).
\end{aligned}
\tag{21}
$$

Using Taylor series once again, similar to (15), to expand $f(z_k)$ around ξ, we get

$$
\begin{aligned}
f(z_k) = f'(\xi)\Big[& \Big(C_2^3 - C_2 C_3\Big)e_k^4 - 2\Big(2C_2^4 - 4C_2^2 C_3 + C_3^2 + C_2 C_4\Big)e_k^5 + \\
& \Big(10C_2^5 - 30C_2^3 C_3 + 12C_2^2 C_4 - 7C_3 C_4 + 3C_2\Big(6C_3^2 - C_5\Big)\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big)\Big].
\end{aligned}
\tag{22}
$$

Replacing (16) and (20) in u_k and v_k of family (1), we have

$$
\begin{aligned}
u_k = & C_2 e_k - \Big(3C_2^2 - 2C_3\Big)e_k^2 + \Big(8C_2^3 - 10C_2 C_3 + 3C_4\Big)e_k^3 + \Big(-20C_2^4 + 37C_2^2 C_3 - \\
& 8C_3^2 - 14C_2 C_4 + 4C_5\Big)e_k^4 + \Big(48C_2^5 - 118C_2^3 C_3 + 51C_2^2 C_4 - 22C_3 C_4 + \\
& C_2\Big(55C_3^2 - 18C_5\Big) + 5C_6\Big)e_k^5 + \mathcal{O}\Big(e_k^6\Big),
\end{aligned}
\tag{23}
$$

$$
\begin{aligned}
v_k = & 1 + C_2 e_k - 2\Big(C_2^2 - C_3\Big)e_k^2 + 3\Big(C_2^3 - 2C_2 C_3 + C_4\Big)e_k^3 + \Big(-3C_2^4 + 11C_2^2 C_3 - \\
& 4C_3^2 - 8C_2 C_4 + 4C_5\Big)e_k^4 + \Big(-10C_2^3 C_3 + 14C_2^2 C_4 + C_2\Big(11C_3^2 - 10C_5\Big) + \\
& 5(-2C_3 C_4 + C_6)\Big)e_k^5 + \mathcal{O}\Big(e_k^6\Big).
\end{aligned}
\tag{24}
$$

Finally, substituting (16) and (21)–(24) in the third step of family (1), we get

$$
\begin{aligned}
x_{k+1} = \xi + & (1 - \alpha - \gamma)C_2\Big(C_2^2 - C_3\Big)e_k^4 + \Big((-4 + 6\alpha - \beta + 5\gamma)C_2^4 + \\
& (8 - 10\alpha + \beta - 9\gamma)C_2^2 C_3 - 2(1 - \alpha - \gamma)C_3^2 - 2(1 - \alpha - \gamma)C_2 C_4\Big)e_k^5 + \\
& \Big((10 - 22\alpha + 9\beta - 14\gamma)C_2^5 - (30 - 53\alpha + 15\beta - 39\gamma)C_2^3 C_3 + \\
& 2(6 - 8\alpha + \beta - 7\gamma)C_2^2 C_4 - 7(1 - \alpha - \gamma)C_3 C_4 + \\
& C_2\Big((18 - 25\alpha + 4\beta - 21\gamma)C_3^2 - 3(1 - \alpha - \gamma)C_5\Big)\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big),
\end{aligned}
\tag{25}
$$

being the error equation

$$
\begin{aligned}
e_{k+1} = {}& (1 - \alpha - \gamma)C_2\Big(C_2^2 - C_3\Big)e_k^4 + \Big((-4 + 6\alpha - \beta + 5\gamma)C_2^4 + \\
& (8 - 10\alpha + \beta - 9\gamma)C_2^2 C_3 - 2(1 - \alpha - \gamma)C_3^2 - 2(1 - \alpha - \gamma)C_2 C_4\Big)e_k^5 + \\
& \Big((10 - 22\alpha + 9\beta - 14\gamma)C_2^5 - (30 - 53\alpha + 15\beta - 39\gamma)C_2^3 C_3 + \\
& 2(6 - 8\alpha + \beta - 7\gamma)C_2^2 C_4 - 7(1 - \alpha - \gamma)C_3 C_4 + \\
& C_2\Big((18 - 25\alpha + 4\beta - 21\gamma)C_3^2 - 3(1 - \alpha - \gamma)C_5\Big)\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big).
\end{aligned}
\tag{26}
$$

To cancel the factors accompanying e_k^4 and e_k^5 in (26), it must be satisfied that $\alpha + \gamma = 1$, $6\alpha - \beta + 5\gamma = 4$ and $10\alpha - \beta + 9\gamma = 8$. It is easy to show that this system of equations has infinite solutions for

$$
\beta = 1 + \alpha \quad \text{and} \quad \gamma = 1 - \alpha,
\tag{27}
$$

where α is a free parameter. Therefore, replacing (27) in (26), we obtain

$$
e_{k+1} = \Big(6C_2^5 - 7C_2^3 C_3 + C_2 C_3^2\Big)e_k^6 + \mathcal{O}\Big(e_k^7\Big),
\tag{28}
$$

and the proof is finished. □

From Theorem 2, it follows that, if we only hold parameter α in (1), the new triparametric family of iterative methods is reduced to an uniparametric family with an order of convergence of six for any real or complex values of the parameters α, β and γ, as long as (27) is satisfied. Therefore, the iterative expression of the new uniparametric family, dependent only on parameter α and which we will call CMT(α) family, is defined as

$$
\begin{cases}
y_k = x_k - \dfrac{f(x_k)}{f'(x_k)}, \\[2mm]
z_k = y_k - \dfrac{f(y_k)}{2f[x_k, y_k] - f'(x_k)}, \\[2mm]
x_{k+1} = z_k - (\alpha + (1 + \alpha)u_k + (1 - \alpha)v_k)\dfrac{f(z_k)}{f'(x_k)},
\end{cases}
\tag{29}
$$

where $u_k = 1 - \dfrac{f[x_k, y_k]}{f'(x_k)}$, $v_k = \dfrac{f'(x_k)}{f[x_k, y_k]}$, and $k = 0, 1, 2, \ldots$

Because of the results obtained with the convergence analysis carried out, from now on we will only work with CMT(α) family of iterative methods and, to select the best members of this family, we will use the complex dynamics tools discussed in Section 3.

3. Complex Dynamics Behavior

This topic refers to the study of the behavior of a rational function associated with an iterative family or method. From the numerical point of view, the dynamical properties of the referred rational function give us important information about its stability and reliability. The parameter spaces of a family of methods, built from the critical points, allow us to understand the performance of the different members of the family, helping us in the election of a particular one. The dynamical planes show the behavior of these particular methods in terms of the basins of attraction of their fixed points, periodic points, etc. A basin of attraction provides us to visually interpret how a method works based on several initial estimates.

In this section, we present the study of the complex dynamics of CMT (α) family given in (29). To do this, we construct a rational operator associated with the family, on a generic low-degree nonlinear polynomial, and we analyze the stability and convergence of the corresponding fixed and critical points. Then, we construct the parameter spaces of the

free critical points and generate dynamical planes of some methods of the family for good and bad values of α, in terms of stability.

3.1. Rational Operator

The rational operator can be built on any nonlinear function; however, we construct this operator on quadratic polynomials, as the criterion of stability or instability of a method applied to these polynomials can be generalized for other nonlinear functions.

Proposition 1. *Let $p(x) = (x - a)(x - b)$ be a generic quadratic polynomial with roots $a, b \in \mathbb{R}$. Therefore, the rational operator $R_\alpha(x)$ associated with CMT(α) family given in (29) and applied on $p(x)$, is*

$$R_\alpha(x) = \frac{x^6 (x^6 + 5x^5 + 12x^4 + 19x^3 + 21x^2 + 14x + \alpha + 5)}{(\alpha + 5)x^6 + 14x^5 + 21x^4 + 19x^3 + 12x^2 + 5x + 1}, \tag{30}$$

with $\alpha \in \mathbb{C}$ an arbitrary parameter. Furthermore, if $\alpha \in \{-77, -1, 1, 5\}$, $R_\alpha(x)$ is simplified as shown

$$R_{-77}(x) = -\frac{x^6 (x^5 + 6x^4 + 18x^3 + 37x^2 + 58x + 72)}{72x^5 + 58x^4 + 37x^3 + 18x^2 + 6x + 1}, \tag{31}$$

$$R_{-1}(x) = \frac{x^6 (x^4 + 3x^3 + 5x^2 + 6x + 4)}{4x^4 + 6x^3 + 5x^2 + 3x + 1}, \tag{32}$$

$$R_1(x) = \frac{x^6 (x^4 + 4x^3 + 7x^2 + 8x + 6)}{6x^4 + 8x^3 + 7x^2 + 4x + 1}, \tag{33}$$

$$R_5(x) = \frac{x^6 (x^4 + 5x^3 + 11x^2 + 14x + 10)}{10x^4 + 14x^3 + 11x^2 + 5x + 1}. \tag{34}$$

Proof. Let $p(x) = (x - a)(x - b)$ be a generic quadratic polynomial with roots $a, b \in \mathbb{R}$. We apply the iterative scheme given in (29) on $p(x)$ and obtain a rational function $A_{p,\alpha}(x)$ which depends on the roots $a, b \in \mathbb{R}$ and a parameter $\alpha \in \mathbb{C}$. Then, if we use Möbius transformation (see in [7–9]) in $A_{p,\alpha}(x)$ with

$$h(w) = \frac{w - a}{w - b},$$

that satisfies $h(\infty) = 1, h(a) = 0$ and $h(b) = \infty$, we get

$$R_\alpha(x) = \left(h \circ A_{p,\alpha} \circ h^{-1} \right)(x) = \frac{x^6 (x^6 + 5x^5 + 12x^4 + 19x^3 + 21x^2 + 14x + \alpha + 5)}{(\alpha + 5)x^6 + 14x^5 + 21x^4 + 19x^3 + 12x^2 + 5x + 1}, \tag{35}$$

which only depends on an arbitrary parameter $\alpha \in \mathbb{C}$. Furthermore, if we factor numerator and denominator of (35), it is easy to show that for $\alpha \in \{-77, -1, 1, 5\}$ some roots coincide and simplify $R_\alpha(x)$, as it is observed in Equations (31)–(34), and the proof is finished. □

From Proposition 1, for four values of α the rational operator $R_\alpha(x)$ is simpler, so there will be fewer fixed and critical points that can improve the stability of the associated methods. This will be seen in Sections 3.2 and 3.3.

3.2. Analysis and Stability of Fixed Points

We calculate the fixed points of the rational operator $R_\alpha(x)$ given in (30) and analyze their stability.

Proposition 2. *The fixed points of $R_\alpha(x)$ are the roots of the equation $R_\alpha(x) = x$. That is, $x = 0$, $x = \infty$ and the following strange fixed points:*

- $ex_1 = 1$ *(if $\alpha \neq -77$), and*

- $ex_i(\alpha)$ that correspond to the 10 roots of polynomial $x^{10} + 6x^9 + 18x^8 + 37x^7 + 58x^6 - (\alpha - 67)x^5 + 58x^4 + 37x^3 + 18x^2 + 6x + 1$, where $i = 2, ..., 11$.

  The total number of different fixed points varies with the value of α:

- If $\alpha \in \mathbb{C}$ and $\alpha \notin \{-77, -1, 1, 5, 307\}$, then $R_\alpha(x)$ has 13 fixed points.
- If $\alpha = -77$, then $ex_1 = 1$ is not a fixed point and $R_\alpha(x)$ has 12 fixed points.
- If $\alpha \in \{-1, 1, 5\}$, then $R_\alpha(x)$ has 11 fixed points.
- If $\alpha = 307$, then $ex_1 = ex_2 = ex_3 = 1$ and $R_\alpha(x)$ has 11 fixed points.

  The pairs of conjugated strange fixed points, satisfying $ex_i = \dfrac{1}{ex_j}$ for $i \neq j$, are ex_2 and ex_3, ex_4 and ex_5, ex_6 and ex_9, ex_7 and ex_8, and ex_{10} and ex_{11}.

From Proposition 2, we establish there is a minimum of 11 and a maximum of 13 fixed points. Of these, 0 and ∞ correspond to the roots of the original quadratic polynomial $p(x)$, and the strange fixed point $ex_1 = 1$ (if $\alpha \neq -77$) corresponds to the divergence of the original method, before Möbius transformation.

Proposition 3. *The stability of the strange fixed point $ex_1 = 1$, $\forall \alpha \in \mathbb{C} \smallsetminus \{-77\}$, verifies:*

(i)　If $\left| \dfrac{384}{77 + \alpha} \right| < 1$, then ex_1 is an attractor.

(ii)　If $\left| \dfrac{384}{77 + \alpha} \right| > 1$, then ex_1 is a repulsor.

(iii)　If $\left| \dfrac{384}{77 + \alpha} \right| = 1$, then ex_1 is parabolic.

ex_1 *is never a superattractor because* $\left| \dfrac{384}{77 + \alpha} \right| \neq 0$. *The superattracting fixed points that satisfy $|R'_\alpha(x)| = 0$ are $x = 0$, $x = \infty$ and the following strange fixed points:*

- ex_4, ex_5 for $\alpha = -0.949874 \pm 0.16946i$,
- ex_6, ex_9 for $\alpha = 2.40285 \pm 1.11088i$, and
- ex_{10}, ex_{11} for $\alpha = 178.653$.

The repulsive fixed points, which always satisfy $|R'_\alpha(x)| > 1$, are the strange fixed points ex_2 and ex_3.

It is clear that 0 and ∞ are always superattracting fixed points, but the stability of the rest of fixed points depends on the values of parameter α. From Proposition 3, there are 6 strange fixed points that can become superattractors for certain values of α. This means that there would be a basin of attraction of the strange fixed point and it could cause the method not to converge to the solution.

Figure 1 shows the stability surface of the strange fixed point ex_1. In this figure, the zones of attraction (yellow surface) and repulsion (gray surface) are observed, being the first one much greater than the second one. Note that for values of α inside disk, ex_1 is a repulsor; and, for off-disk values of α, ex_1 is an attractor. Therefore, it is in our interest to always work inside the disk because the strange fixed point $ex_1 = 1$ comes from the divergence of the original method and, therefore, it is better for the performance of the iterative method that the divergence is repulsive.

From Proposition 2, the study of the stability of strange fixed points is reduced by a half. This is because each pair of conjugated strange fixed points exhibits the same stability characteristics. Furthermore, due to Proposition 3, ex_2 and ex_3 are always repulsors regardless of the value of α. Thus, Figure 2 shows the stability surfaces of the remaining 8 strange fixed points, which can be attracting or repulsive depending on the value of α, for analysis.
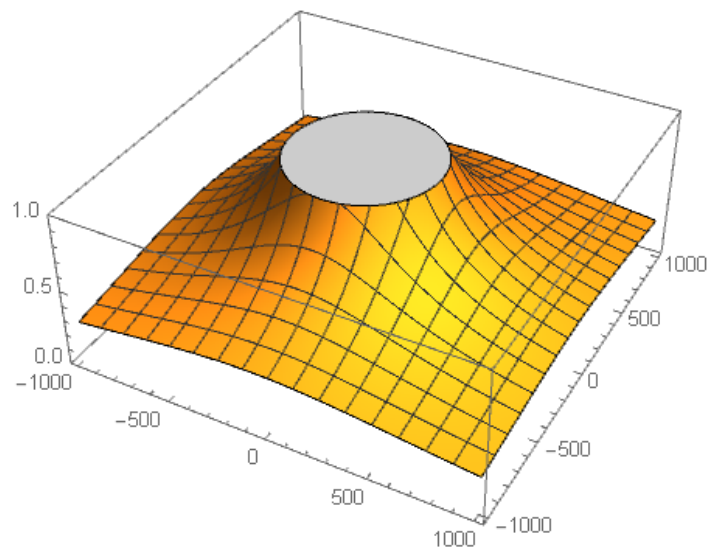
Figure 1. Stability surface of $ex_1 = 1$ (in gray color, the complex area where the fixed point is repulsive, being attracting in the rest).

3.3. Analysis of Critical Points

We calculate the critical points of the rational operator $R_\alpha(x)$ given in (30).

Proposition 4. *The critical points of $R_\alpha(x)$ are the roots of the equation $R'_\alpha(x) = 0$. That is, $x = 0$, $x = \infty$ and the following free critical points:*

- $cr_1 = -1$,
- $cr_2 = -i$,
- $cr_3 = i$, *and*
- $cr_i(\alpha)$ *that correspond to the 6 roots of polynomial $(6\alpha + 30)x^6 + (\alpha + 103)x^5 + (2\alpha + 206)x^4 + (-6\alpha + 246)x^3 + (2\alpha + 206)x^2 + (\alpha + 103)x + 6\alpha + 30$, where $i = 4, ..., 9$.*

The total number of different critical points varies with the value of α:

- *If $\alpha \in \mathbb{C}$ and $\alpha \notin \{-77, -5, -1, 1, 5\}$, then $R_\alpha(x)$ has 11 critical points.*
- *If $\alpha \in \{-77, -5, -1\}$, then $R_\alpha(x)$ is simplified or reduced and has 9 critical points.*
- *If $\alpha \in \{1, 5\}$, then $R_\alpha(x)$ is simplified and has 7 critical points.*

The pairs of conjugated free critical points, satisfying $cr_i = \dfrac{1}{cr_j}$ for $i \neq j$, are cr_2 and cr_3, cr_4 and cr_5, cr_6 and cr_7, and cr_8 and cr_9.

From Proposition 4, we establish there is a minimum of 7 and a maximum of 11 critical points. Of these, 0 and ∞ correspond to the roots of the original quadratic polynomial $p(x)$. The free critical points $cr_1 = -1$, $cr_2 = -i$, and $cr_3 = i$ are pre-images of the strange fixed point $ex_1 = 1$. Therefore, the stability of cr_1, cr_2, and cr_3 will correspond to the stability of ex_1 (see Section 3.2). Moreover, the dynamical study of the free critical points is reduced by a half because each pair of conjugated free critical points presents the same stability characteristics. This will be seen in Section 3.4.

(**a**) ex_4 and ex_5

(**b**) ex_6 and ex_9

(**c**) ex_7 and ex_8

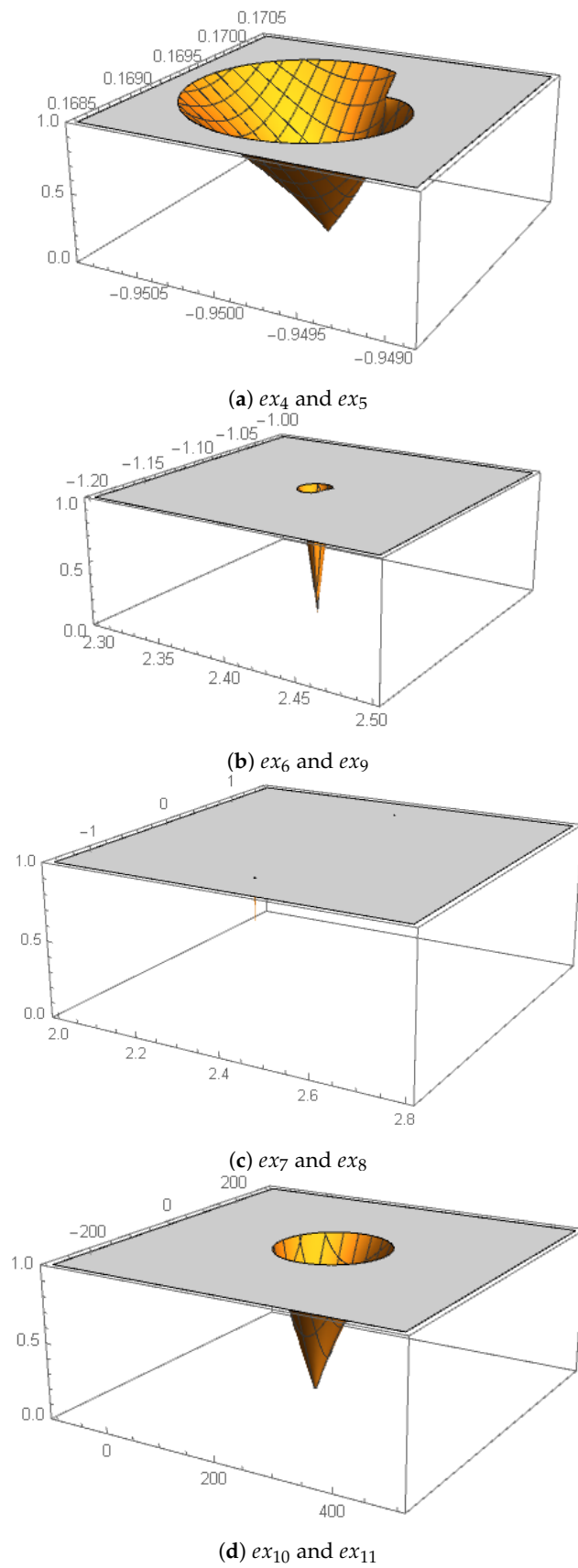(**d**) ex_{10} and ex_{11}

Figure 2. Stability surfaces of 8 strange fixed points (in gray color, the complex area where each fixed point is repulsive, being attracting in the rest).

3.4. Parameter Spaces

The dynamical behavior of operator $R_\alpha(x)$ depends on the values of parameter α. The parameter space is defined as a mesh in the complex plane where each point of this mesh corresponds to a different value of α. Its graphical representation shows the convergence analysis of a method of CMT (α) family associated with this α using one of the free critical points $cr(\alpha)$ given in Proposition 4 as initial estimate. The resulting graphic is made in Matlab R2020a programming package with a resolution of 1000×1000 pixels. If a method converges to any of the roots starting from $cr(\alpha)$ in a maximum of 80 iterations with a tolerance of 10^{-3}, the pixel is colored red; in other cases, the pixel is colored black.

Each value of α that belongs to the same connected component of the parameter space results in subsets of schemas with similar dynamical behavior. Therefore, it is interesting to find regions of the parameter space as stable as possible (red regions), because these values of α will give us the best members of the family in terms of numerical stability.

CMT(α) family has a maximum of 9 free critical points. Of these, cr_1, cr_2, and cr_3 have the same parameter space which corresponds to the stability surface of ex_1 (see Figure 1), because they are pre-images of this point. The remaining free critical points, cr_4 to cr_9, are conjugated in pairs (see Proposition 4), which gives rise to 3 different parameter spaces. These parameter spaces, named P_1 (for $x = cr_4, cr_5$), P_2 (for $x = cr_6, cr_7$), and P_3 (for $x = cr_8, cr_9$), are shown in Figure 3.

From Figure 3b,c, we observe that the parameter spaces P_2 and P_3 have similar characteristics; then, we can select any of them for analysis.

On the one hand, if we choose values of α inside the stability regions (red regions) of the parameter spaces, for example, $\alpha = -1, 0, 1$, the methods associated with these parameters will show good dynamical behavior in terms of numerical stability. Furthermore, note that these particular values of α simplify the iterative scheme of CMT(α) family given in (29) by canceling a term in its third step. This is especially useful to improve the computational efficiency of the associated method because the processing times required to reach the solution are reduced (see Section 4).

On the other hand, if we choose values of α outside the stability regions (black regions) of the parameter spaces, for example $\alpha = -300, 200, 400$, the methods associated with these parameters will show poor dynamical behavior in terms of numerical stability.

The methods associated with the values of α treated above are discussed in Section 3.5.

3.5. Dynamical Planes

We begin this section by presenting how we generate a dynamical plane that will allow us to see the stability of a method for a specific value of α. This is defined as a mesh in the complex plane where each point of this mesh corresponds to a different value of the initial estimate x_0. Its graphical representation shows the convergence of the method to any of the roots starting from x_0 with a maximum of 50 iterations and a tolerance of 10^{-3}. Fixed points are illustrated with a white circle "○", critical points with a white square "□" and attractors with a white asterisk "∗". Moreover, the basins of attraction are depicted in different colors. The resulting graphic is made in Matlab R2020a with a resolution of 1000×1000 pixels.

Here, we study the stability of some CMT(α) family methods through the use of dynamical planes. We will consider the methods proposed in Section 3.4 for values of α inside and outside the stability regions of the parameter spaces.

On the one hand, examples of methods inside the stability region are given for $\alpha = -1, 0, 1$. Their dynamical planes with some convergence orbits in yellow are shown in Figure 4. Note that all three methods present only two basins of attraction associated with the roots: the basin of 0 colored in orange and the basin of ∞ colored in blue. Furthermore, there are no black areas of non-convergence to the solution. Consequently, these methods show good dynamical behavior: they are very stable. Of these methods, the best member of CMT(α) family is for $\alpha = 1$, as it has fewer strange fixed points and free critical points.
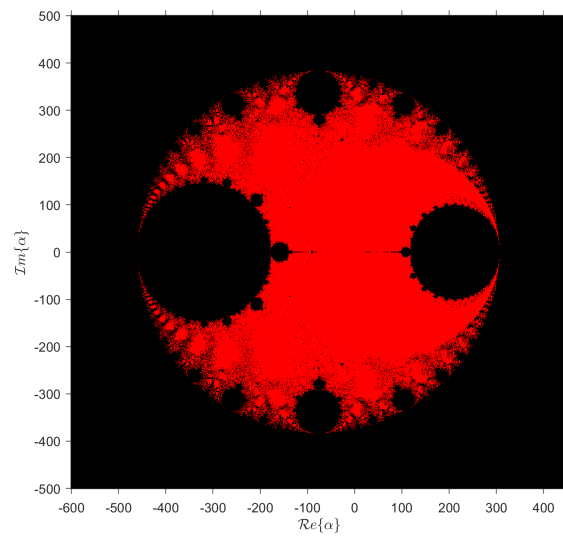
(a) P_1 for $x = cr_4, cr_5$
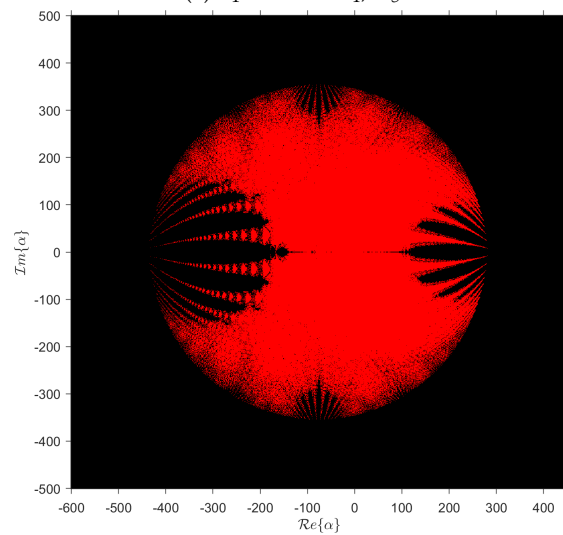
(b) P_2 for $x = cr_6, cr_7$
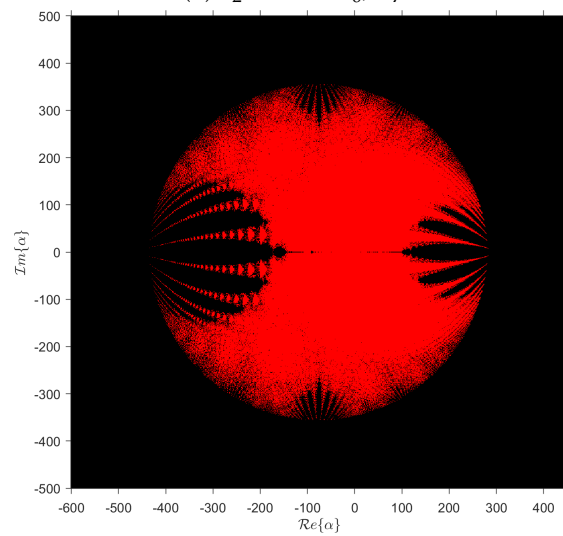
(c) P_3 for $x = cr_8, cr_9$

Figure 3. Parameter spaces of free critical points (in red color, the complex area where the corresponding critical point converges to 0 or ∞, that is, the stability region).

(**a**) $\alpha = -1$

(**b**) $\alpha = 0$

(**c**) $\alpha = 1$
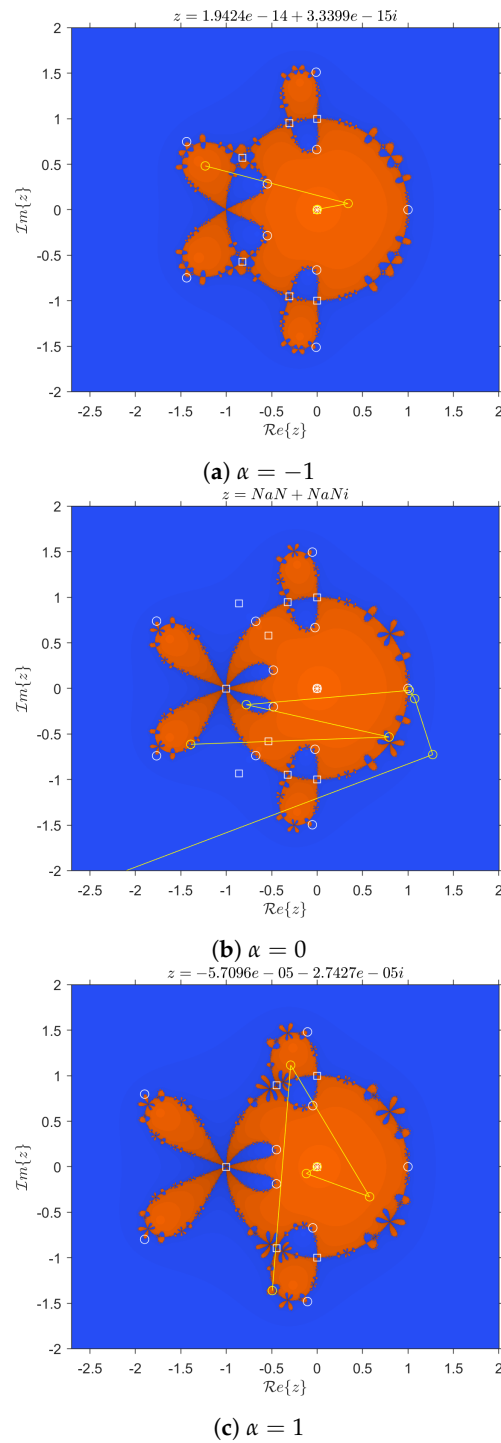
Figure 4. Dynamical planes for methods inside the stability region (basin of attraction of 0 in orange color; in blue color, the basin of ∞).

On the other hand, examples of methods outside the stability region are given for $\alpha = -300$, 200, 400. Their dynamical planes with some convergence orbits in yellow are shown in Figure 5. Note that all three methods present more than two basins of attraction, that is, there are other basins of attraction that do not correspond to the roots. The basins of 0 and ∞ are colored in orange and blue, respectively, and the other basins are colored in black, red, and green. Figure 5a shows the convergence to an attracting periodic orbit of period 2. Figure 5b,c shows the convergence to an attracting strange fixed point. Furthermore, let us remark that in the three figures the basin of 0 is very small, due to the presence of the other basins of attraction, which reduces the chances of convergence to the solution. Likewise,

there are black areas of slow convergence of the methods. Consequently, these methods have poor dynamical behavior: they are unstable.
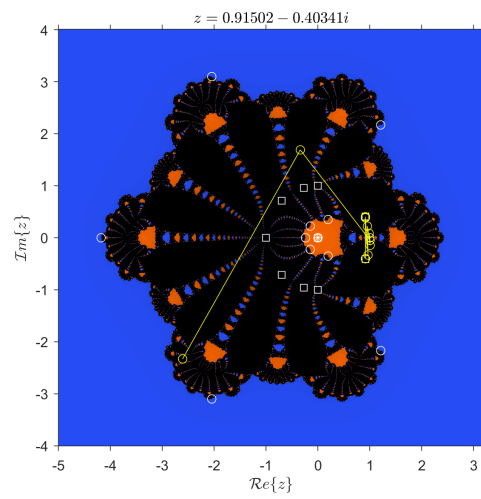
(**a**) $\alpha = -300$
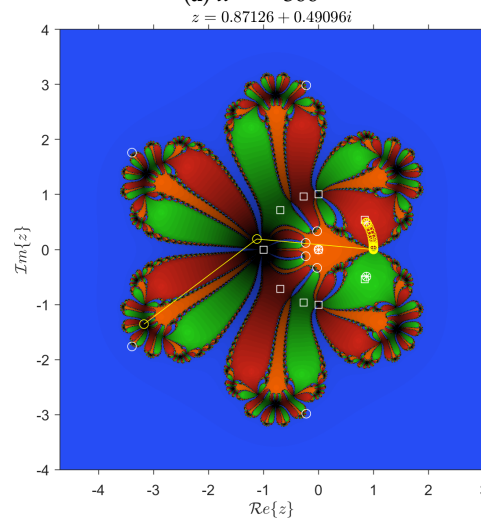
(**b**) $\alpha = 200$
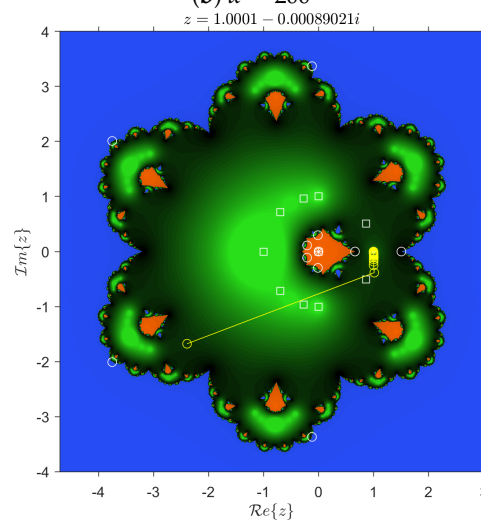
(**c**) $\alpha = 400$

Figure 5. Dynamical planes for methods outside the stability region (basin of attraction of 0 in orange color; in blue color, the basin of ∞; in green or red color, the basin of attracting strange fixed points).

4. Numerical Results

Here, we perform several numerical tests in order to check the theoretical convergence and stability results of CMT(α) family obtained in previous sections. To do this, we use some stable and unstable methods of (29). These methods are applied on five nonlinear test functions, whose expressions and corresponding roots are

$$f_1(x) = \sin(x) - x^2 + 1, \quad \xi \approx -0.6367326508,$$
$$f_2(x) = \cos(x) - x\exp(x) + x^2, \quad \xi \approx 0.6391540963,$$
$$f_3(x) = x^3 + 4x^2 - 10, \quad \xi \approx 1.3652300134,$$
$$f_4(x) = \sqrt{x^2 + 2x + 5} - 2\sin(x) - x^2 + 3, \quad \xi \approx 2.3319676559,$$
$$f_5(x) = \sqrt{x^4} + \sin\left(\frac{\pi}{x^2}\right) - \frac{3}{16}, \quad \xi \approx -0.9059869793.$$

Thus, we performed two experiments. In a first experiment, we carried out a efficiency analysis of CMT(α) family through a comparative study between one of its stable methods and five different methods given in the literature: Newton of order 2, Ostrowski of order 4, and three other methods of order 6 proposed by Alzahrani et al. in [10] (ABA), Chun and Ham in [11] (CH), and Amat et al. in [12] (AHR). In a second experiment, we carried out a stability analysis of CMT (α) family using six of its methods obtained with three good and three bad values of parameter α, in terms of stability.

In the development of the numerical tests we start the iterations with different initial estimates: close ($x_0 \approx \xi$), far ($x_0 \approx 10\xi$), and very far ($x_0 \approx 100\xi$) to the root ξ, respectively. This allows us to measure, to some extent, how demanding the methods are relative to the initial estimation for finding a solution.

The calculations are developed in Matlab R2020a programming package using variable precision arithmetics with 200 digits of mantissa. For each method, we analyze the number of iterations (iter) required to converge to the solution, so that the stopping criteria $|x_{k+1} - x_k| < 10^{-100}$ or $|f(x_{k+1})| < 10^{-100}$ are satisfied. Note that $|x_{k+1} - x_k|$ represents the error estimation between two consecutive iterations and $|f(x_{k+1})|$ is the residual error of the nonlinear test function. This stopping criterium does not need the exact solution, on the contrary of absolute error, and differs from recent ones as CESTAC (see in [13]) in the absence of additional calculations or functional evaluations, as $f(x_{k+1})$ is needed for the following iteration and its absolute values is an efficient control element of the proximity to the exact root, where f is zero. Indeed, although a precision of one hundred exact digits is not usually necessary in the applications, we employ this value in the stopping criterium as it is useful to check the robustness and effectiveness of the numerical methods.

To check the theoretical order of convergence (p), we calculate the approximate computational order of convergence (ACOC) given by Cordero and Torregrosa in [14]. In the numerical results presented below, if the ACOC vector inputs do not stabilize their values throughout the iterative process, it is marked as "-"; and, if any of the methods used does not reach convergence in a maximum of 50 iterations, it is marked as "nc".

To illustrate the computational efficiency of each used method, the processing time (tcpu) in seconds required by the iterative scheme to converge to the solution is measured. This value is determined as the arithmetic mean of 10 runs of the method.

4.1. First Experiment: Efficiency Analysis of CMT (α) Family

In this experiment, we carried out a comparative study between a stable method of CMT (α) family and the methods of Newton, Ostrowski, ABA, CH and AHR, in order to contrast their numerical performances in nonlinear equations. We consider as a stable member of CMT (α) family the method associated with $\alpha = 1$, that is, CMT(1).

Thereby, in Tables 1–3 we show the numerical results of the six known methods, considering close, far, and very far initial estimates. Furthermore, in Figure 6 we show graphics that summarize these results for the number of iterations (iter) and the processing time (tcpu).

Therefore, from the results of the first experiment we conclude that CMT(α) family has an excellent numerical performance considering a stable member ($\alpha = 1$) as a representative. This conclusion has been made based on the following aspects from Tables 1–3: CMT (1) method has the lowest error and lowest number of iterations (iter). However, the mean of the execution time (tcpu) varies according to the nonlinear test function used and the inherent complexity that the iterative scheme of the method presents on the nonlinear function. In several cases, the tcpu of the CMT (1) method is significantly lower than the 6th order ABA, CH and AHR methods. The theoretical convergence order is also verified by the ACOC, which is close to 6.

Table 1. Numerical performance of iterative methods in nonlinear equations for x_0 **close to** ξ.

Function	Method	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
f_1	CMT(1)	7.6395×10^{-19}	1.8769×10^{-110}	3	5.5148	0.1257
$x_0 = -1.6$	Newton	3.2063×10^{-84}	7.2243×10^{-168}	8	2	0.1225
	Ostrowski	3.6277×10^{-39}	2.1775×10^{-155}	4	3.9988	0.1036
	ABA	6.3941×10^{-19}	5.2542×10^{-111}	3	5.5472	0.1201
	CH	3.9619×10^{-19}	3.095×10^{-112}	3	5.5336	0.1173
	AHR	6.9779×10^{-86}	0	4	5.9989	0.1381
f_2	CMT(1)	1.1915×10^{-19}	3.2336×10^{-114}	4	6.0717	0.2913
$x_0 = -0.4$	Newton	6.977×10^{-101}	9.2573×10^{-201}	10	2	0.2747
	Ostrowski	3.6009×10^{-28}	5.8295×10^{-111}	4	3.9993	0.1899
	ABA	6.593×10^{-46}	0	4	6.0055	0.4278
	CH	4.0133×10^{-50}	6.8135×10^{-208}	5	5.9951	0.5636
	AHR	1.4561×10^{-73}	0	10	5.9991	0.7038
f_3	CMT(1)	5.868×10^{-64}	0	7	5.9957	0.4654
$x_0 = 0.4$	Newton	3.2665×10^{-83}	8.6382×10^{-165}	10	2	0.2818
	Ostrowski	1.3665×10^{-51}	5.077×10^{-204}	5	3.9999	0.1682
	ABA	2.5625×10^{-27}	4.3729×10^{-160}	5	5.8933	0.224
	CH	2.4971×10^{-24}	4.0266×10^{-142}	9	5.8498	0.4374
	AHR	2.1589×10^{-36}	0	12	5.9521	0.5912
f_4	CMT(1)	1.2572×10^{-32}	3.2096×10^{-195}	3	5.717	0.6075
$x_0 = 1.3$	Newton	7.2803×10^{-95}	1.2821×10^{-189}	7	2	0.4947
	Ostrowski	1.0395×10^{-64}	1.5574×10^{-207}	4	4	0.535
	ABA	4.7735×10^{-26}	8.9685×10^{-156}	3	5.9419	0.598
	CH	1.6112×10^{-32}	1.7919×10^{-194}	3	5.6961	0.6046
	AHR	3.0816×10^{-22}	1.1423×10^{-131}	3	5.6812	0.4497
f_5	CMT(1)	2.5535×10^{-53}	6.4242×10^{-207}	6	5.9132	1.2222
$x_0 = -1.9$	Newton	3.4167×10^{-84}	8.1562×10^{-167}	8	2	0.6295
	Ostrowski	4.1408×10^{-38}	2.4627×10^{-142}	4	4.0146	0.5521
	ABA	5.637×10^{-65}	1.9467×10^{-208}	5	6.0107	0.9561
	CH	1.0828×10^{-43}	1.0707×10^{-207}	6	6.2212	1.1314
	AHR	5.6988×10^{-26}	4.6285×10^{-106}	4	5.7855	0.5939

4.2. Second Experiment: Stability Analysis of CMT(α) Family

In this experiment, we carried out a stability analysis of CMT (α) family considering some values of α inside the stability regions of the parameter spaces ($\alpha = -1,\ 0,\ 1$) and outside of them ($\alpha = -300,\ 200,\ 400$).

Thus, in Tables 4–9 we show the numerical performance of iterative methods associated with these values of α for close, far, and very far initial estimations. The results for $\alpha = 1$ were already presented in the first experiment; however, these are presented again due to the different conditions in which each experiment was performed.

On the one hand, from Tables 4–6 we observe that the methods associated with $\alpha = -1,\ 0,\ 1$ always converge to the solution, although the number of iterations (iter) needed differs for any initial estimate and nonlinear test function. Thus, in estimations close to the root, the methods converge to ξ with a minimum iter of 3 and a maximum of 7. When the initial guess is far from the root, they converge to ξ with a minimum iter of 4 and a maximum of 22. When the starting estimations are very far from the root, the iterative schemes converge to ξ with a minimum iter of 6 and a maximum of 37.

Table 2. Numerical performance of iterative methods in nonlinear equations for x_0 **far from** ξ.

Function	Method	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
f_1	CMT(1)	4.3721×10^{-23}	6.595×10^{-136}	4	5.7093	0.163
$x_0 = -6$	Newton	4.549×10^{-85}	1.4542×10^{-169}	10	2	0.1527
	Ostrowski	7.5454×10^{-40}	4.0753×10^{-158}	5	3.9989	0.1487
	ABA	6.5662×10^{-25}	6.1621×10^{-147}	4	5.775	0.1607
	CH	9.4464×10^{-24}	5.6868×10^{-140}	4	5.7326	0.1811
	AHR	9.9786×10^{-85}	0	5	5.9988	0.1578
f_2	CMT(1)	6.0086×10^{-60}	0	16	5.9975	0.9939
$x_0 = -6$	Newton	2.9103×10^{-57}	1.6107×10^{-113}	12	2	0.2714
	Ostrowski	1.7318×10^{-82}	6.8135×10^{-208}	8	4	0.3234
	ABA	2.9737×10^{-18}	6.4713×10^{-106}	10	-	0.6234
	CH	4.8167×10^{-51}	0	14	5.9955	0.8618
	AHR	1.0711×10^{-58}	0	6	5.9971	0.3006
f_3	CMT(1)	4.2145×10^{-24}	1.1268×10^{-140}	10	5.8416	0.4353
$x_0 = -14$	Newton	nc	nc	nc	nc	nc
	Ostrowski	2.3325×10^{-76}	0	37	4	0.9868
	ABA	9.1479×10^{-18}	9.0509×10^{-103}	24	6.2542	1.1023
	CH	5.0027×10^{-98}	0	17	5.9997	0.7088
	AHR	nc	nc	nc	nc	nc
f_4	CMT(1)	4.6353×10^{-98}	2.3361×10^{-207}	5	5.9995	0.978
$x_0 = -23$	Newton	9.6577×10^{-79}	1.3216×10^{-156}	10	2	0.683
	Ostrowski	8.1672×10^{-31}	5.8293×10^{-122}	5	3.9956	0.6646
	ABA	1.3364×10^{-69}	2.3361×10^{-207}	5	5.9961	0.9691
	CH	4.543×10^{-99}	2.3361×10^{-207}	5	5.9996	0.9597
	AHR	1.7793×10^{-56}	2.3361×10^{-207}	5	5.9898	0.6951
f_5	CMT(1)	3.9117×10^{-41}	9.6363×10^{-207}	5	6.1766	0.9564
$x_0 = -9$	Newton	1.2423×10^{-55}	1.9722×10^{-109}	9	2	0.615
	Ostrowski	2.9225×10^{-29}	1.7446×10^{-112}	5	3.9821	0.6514
	ABA	2.0254×10^{-31}	5.3498×10^{-181}	5	5.5153	0.9702
	CH	6.524×10^{-29}	3.5709×10^{-159}	6	6.2558	1.1451
	AHR	1.6141×10^{-41}	9.7687×10^{-148}	12	5.8222	1.592

Table 3. Numerical performance of iterative methods in nonlinear equations for x_0 **very far from** ξ.

Function	Method	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
f_1	CMT(1)	6.8586×10^{-80}	0	6	5.9981	0.2413
$x_0 = -60$	Newton	3.1826×10^{-73}	7.1179×10^{-146}	13	2	0.2003
	Ostrowski	1.2267×10^{-100}	0	7	4	0.1793
	ABA	1.3417×10^{-77}	0	6	5.9978	0.273
	CH	1.4971×10^{-82}	0	6	5.9984	0.2776
	AHR	8.7686×10^{-61}	3.8934×10^{-208}	7	5.992	0.2246
f_2	CMT(1)	5.9893×10^{-27}	5.2167×10^{-158}	6	6.0379	0.3503
$x_0 = -60$	Newton	1.6537×10^{-59}	5.201×10^{-118}	15	2	0.3125
	Ostrowski	8.0088×10^{-72}	0	8	4	0.2956
	ABA	2.9305×10^{-56}	0	10	6.0024	0.5679
	CH	8.4413×10^{-48}	0	7	5.994	0.399
	AHR	6.4484×10^{-60}	0	7	5.9974	0.318
f_3	CMT(1)	3.7145×10^{-76}	0	13	5.9983	0.6398
$x_0 = -140$	Newton	nc	nc	nc	nc	nc
	Ostrowski	6.9267×10^{-37}	3.3507×10^{-145}	49	3.999	1.216
	ABA	7.5885×10^{-54}	0	11	5.9907	0.4246
	CH	4.8283×10^{-28}	2.1045×10^{-164}	21	5.8989	0.8005
	AHR	3.4494×10^{-58}	6.2295×10^{-207}	12	5.9928	0.3997
f_4	CMT(1)	9.2602×10^{-68}	2.3361×10^{-207}	6	5.9954	1.0547
$x_0 = -230$	Newton	8.9492×10^{-96}	1.1348×10^{-190}	14	2	0.8454
	Ostrowski	7.8874×10^{-37}	5.0705×10^{-146}	7	3.9985	0.8196
	ABA	2.5587×10^{-21}	9.9754×10^{-126}	6	6.2382	1.0537
	CH	2.2055×10^{-60}	2.3361×10^{-207}	6	6.0079	1.0555
	AHR	nc	nc	nc	nc	nc
f_5	CMT(1)	2.8545×10^{-38}	1.0707×10^{-207}	6	6.2665	1.0249
$x_0 = -90$	Newton	9.6307×10^{-58}	6.4804×10^{-114}	12	2	0.7181
	Ostrowski	6.1241×10^{-52}	6.9183×10^{-202}	8	3.9999	0.9291
	ABA	1.2306×10^{-20}	2.1729×10^{-114}	6	6.8491	1.0378
	CH	3.4946×10^{-26}	1.6995×10^{-147}	6	5.7567	1.0301
	AHR	8.5778×10^{-51}	1.0901×10^{-182}	25	5.902	3.0345

On the other hand, from the results shown in Tables 7–9, we see that the methods associated with $\alpha = -300, 200, 400$ do not always converge to the solution, confirming the conclusions obtained in the dynamical analysis. The convergence highly depends on the initial estimation and the nonlinear test function used. Thus, for estimations close to the root, these methods do not converge to the solution in up to 2 test functions. Moreover, for estimations far and very far from the root, they do not converge to the solution even for any function.

Consequently, we conclude that the methods for $\alpha = -1$, 0, 1 are stable, have the lowest processing times (tcpu), and always converge to the solution for any initial estimate and nonlinear test function used. The methods for $\alpha = -300$, 200, 400 are unstable, chaotic, have the highest tcpu, and tend not to converge to the solution according to the initial estimate and the nonlinear test function used. With this, the theoretical results obtained in previous sections about the dynamical behavior of CMT(α) family are verified.
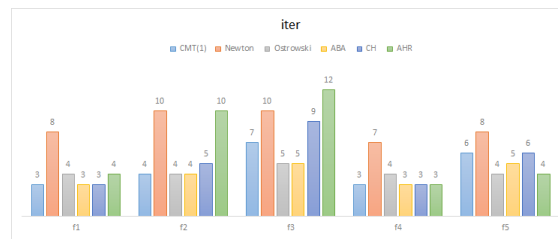
(**a**) Number of iterations for x_0 close to ξ
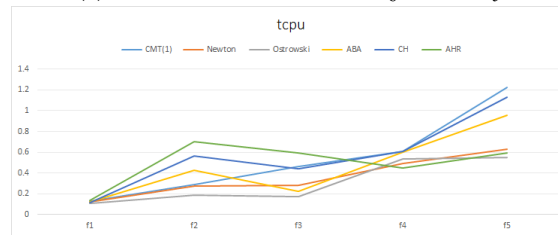
(**b**) Processing time for x_0 close to ξ
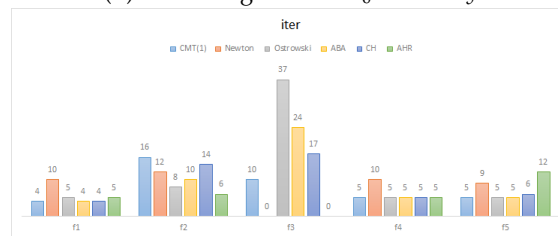
(**c**) Number of iterations for x_0 far from ξ
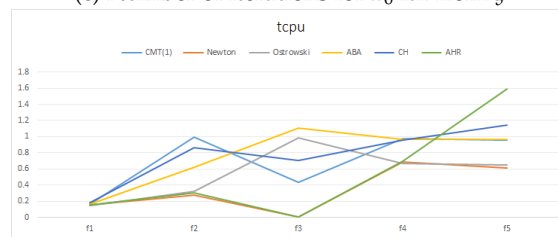
(**d**) Processing time for x_0 far from ξ
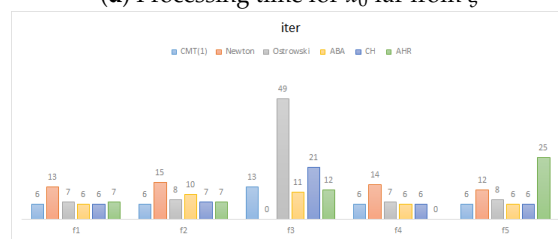
(**e**) Number of iterations for x_0 very far from ξ
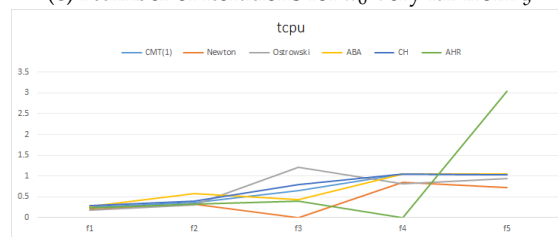
(**f**) Processing time for x_0 very far from ξ

Figure 6. Numerical results of the first experiment.

Table 4. Numerical performance of CMT (-1) method in nonlinear equations.

Function	x_0	$\lvert x_{k+1} - x_k \rvert$	$\lvert f(x_{k+1}) \rvert$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	1.8646×10^{-19}	2.7591×10^{-114}	3	5.5559	0.1216
f_2	-0.4	1.3898×10^{-46}	0	4	6.0038	0.2775
f_3	0.4	9.0583×10^{-50}	0	5	5.9873	0.2321
f_4	1.3	1.9771×10^{-32}	7.3778×10^{-194}	3	5.6791	0.6628
f_5	-1.9	4.057×10^{-47}	1.606×10^{-206}	6	6.0586	1.2462
	Far from ξ					
f_1	-6	1.4965×10^{-24}	7.3749×10^{-145}	4	5.7594	0.1606
f_2	-6	7.3835×10^{-26}	1.1396×10^{-151}	14	-	0.8807
f_3	-14	1.009×10^{-18}	1.3833×10^{-108}	22	5.7241	0.9937
f_4	-23	3.2059×10^{-100}	2.3361×10^{-207}	5	5.9996	1.0545
f_5	-9	4.5305×10^{-85}	1.168×10^{-207}	7	6.0034	1.446
	Very far from ξ					
f_1	-60	1.264×10^{-85}	0	6	5.9988	0.2385
f_2	-60	8.3236×10^{-19}	2.3391×10^{-109}	9	6.0055	0.5682
f_3	-140	6.8807×10^{-19}	1.3913×10^{-109}	10	5.7297	0.4723
f_4	-230	1.1069×10^{-48}	2.3361×10^{-207}	6	6.0195	1.2992
f_5	-90	2.3226×10^{-65}	1.168×10^{-207}	6	5.9808	1.3969

Table 5. Numerical performance of CMT (0) method in nonlinear equations.

Function	x_0	$\lvert x_{k+1} - x_k \rvert$	$\lvert f(x_{k+1}) \rvert$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	3.9254×10^{-19}	2.9279×10^{-112}	3	5.5334	0.1219
f_2	-0.4	1.0637×10^{-28}	1.328×10^{-168}	4	6.0263	0.2689
f_3	0.4	4.828×10^{-30}	2.1036×10^{-176}	6	5.9174	0.2482
f_4	1.3	1.6112×10^{-32}	1.7919×10^{-194}	3	5.6961	0.6771
f_5	-1.9	3.0345×10^{-27}	1.8831×10^{-154}	6	6.5022	1.2896
	Far from ξ					
f_1	-6	8.7386×10^{-24}	3.5638×10^{-140}	4	5.7334	0.1602
f_2	-6	6.7903×10^{-26}	8.9867×10^{-152}	9	-	0.6206
f_3	-14	8.1206×10^{-24}	4.7631×10^{-139}	11	5.8407	0.491
f_4	-23	4.2612×10^{-99}	2.3361×10^{-207}	5	5.9996	1.0585
f_5	-9	4.1362×10^{-41}	2.3361×10^{-207}	6	5.9265	1.2619
	Very far from ξ					
f_1	-60	1.1445×10^{-82}	0	6	5.9985	0.2395
f_2	-60	3.277×10^{-54}	0	7	5.9966	0.4971
f_3	-140	3.695×10^{-64}	0	37	5.9959	1.6934
f_4	-230	5.2233×10^{-59}	2.3361×10^{-207}	6	6.0088	1.2644
f_5	-90	8.2696×10^{-19}	2.984×10^{-103}	6	5.5602	1.2865

Table 6. Numerical performance of CMT (1) method in nonlinear equations.

Function	x_0	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	7.6395×10^{-19}	1.8769×10^{-110}	3	5.5148	0.124
f_2	-0.4	1.1915×10^{-19}	3.2336×10^{-114}	4	6.0717	0.2474
f_3	0.4	5.868×10^{-64}	0	7	5.9957	0.3128
f_4	1.3	1.2572×10^{-32}	3.2096×10^{-195}	3	5.717	0.7052
f_5	-1.9	2.5535×10^{-53}	6.4242×10^{-207}	6	5.9132	1.3006
	Far from ξ					
f_1	-6	4.3721×10^{-23}	6.595×10^{-136}	4	5.7093	0.1619
f_2	-6	6.0086×10^{-60}	0	16	5.9975	1.0008
f_3	-14	4.2145×10^{-24}	1.1268×10^{-140}	10	5.8416	0.446
f_4	-23	4.6353×10^{-98}	2.3361×10^{-207}	5	5.9995	1.0401
f_5	-9	3.9117×10^{-41}	9.6363×10^{-207}	5	6.1766	1.0393
	Very far from ξ					
f_1	-60	6.8586×10^{-80}	0	6	5.9981	0.2654
f_2	-60	5.9893×10^{-27}	5.2167×10^{-158}	6	6.0379	0.3777
f_3	-140	3.7145×10^{-76}	0	13	5.9983	0.5816
f_4	-230	9.2602×10^{-68}	2.3361×10^{-207}	6	5.9954	1.2349
f_5	-90	2.8545×10^{-38}	1.0707×10^{-207}	6	6.2665	1.2801

Table 7. Numerical performance of CMT (-300) method in nonlinear equations.

Function	x_0	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	1.454×10^{-49}	3.8934×10^{-208}	4	6.0127	0.1743
f_2	-0.4	7.7×10^{-75}	0	40	6.0006	2.5385
f_3	0.4	nc	nc	nc	nc	nc
f_4	1.3	4.1603×10^{-29}	3.3384×10^{-172}	3	5.3365	0.621
f_5	-1.9	1.6341×10^{-58}	2.5794×10^{-206}	5	5.7418	1.1787
	Far from ξ					
f_1	-6	nc	nc	nc	nc	nc
f_2	-6	nc	nc	nc	nc	nc
f_3	-14	3.9697×10^{-26}	4.0419×10^{-151}	7	6.0709	0.328
f_4	-23	nc	nc	nc	nc	nc
f_5	-9	2.8218×10^{-76}	6.0348×10^{-207}	8	5.9788	1.5886
	Very far from ξ					
f_1	-60	4.4607×10^{-32}	3.3463×10^{-187}	9	6.0453	0.3717
f_2	-60	nc	nc	nc	nc	nc
f_3	-140	1.7723×10^{-57}	0	21	6.0044	0.9822
f_4	-230	1.249×10^{-29}	2.4449×10^{-175}	39	5.1386	7.4938
f_5	-90	6.6349×10^{-43}	4.8668×10^{-209}	22	6.0131	4.4952

Table 8. Numerical performance of CMT (200) method in nonlinear equations.

Function	x_0	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	2.1496×10^{-56}	0	4	5.9921	0.1499
f_2	-0.4	nc	nc	nc	nc	nc
f_3	0.4	nc	nc	nc	nc	nc
f_4	1.3	4.0045×10^{-33}	1.6998×10^{-196}	3	5.3325	0.6325
f_5	-1.9	1.3149×10^{-70}	9.6363×10^{-207}	4	6.0496	0.8213
	Far from ξ					
f_1	-6	6.1599×10^{-40}	3.8934×10^{-208}	7	5.9673	0.2711
f_2	-6	nc	nc	nc	nc	nc
f_3	-14	nc	nc	nc	nc	nc
f_4	-23	4.3946×10^{-33}	2.9689×10^{-196}	7	5.339	1.4742
f_5	-9	8.369×10^{-63}	2.9162×10^{-205}	11	5.964	2.0915
	Very far from ξ					
f_1	-60	1.9877×10^{-20}	1.8239×10^{-118}	14	5.7565	0.5598
f_2	-60	nc	nc	nc	nc	nc
f_3	-140	nc	nc	nc	nc	nc
f_4	-230	2.7541×10^{-49}	1.5574×10^{-207}	15	5.9586	3.1228
f_5	-90	7.8278×10^{-51}	9.6363×10^{-207}	15	6.1663	3.2771

Table 9. Numerical performance of CMT (400) method in nonlinear equations.

Function	x_0	$|x_{k+1} - x_k|$	$|f(x_{k+1})|$	iter	ACOC	tcpu
	Close to ξ					
f_1	-1.6	2.9103×10^{-44}	0	4	5.9805	0.1439
f_2	-0.4	nc	nc	nc	nc	nc
f_3	0.4	nc	nc	nc	nc	nc
f_4	1.3	1.139×10^{-35}	1.5574×10^{-207}	3	5.2494	0.6218
f_5	-1.9	5.8131×10^{-53}	3.1147×10^{-207}	4	5.754	0.8023
	Far from ξ					
f_1	-6	nc	nc	nc	nc	nc
f_2	-6	nc	nc	nc	nc	nc
f_3	-14	nc	nc	nc	nc	nc
f_4	-23	nc	nc	nc	nc	nc
f_5	-9	nc	nc	nc	nc	nc
	Very far from ξ					
f_1	-60	nc	nc	nc	nc	nc
f_2	-60	nc	nc	nc	nc	nc
f_3	-140	nc	nc	nc	nc	nc
f_4	-230	nc	nc	nc	nc	nc
f_5	-90	nc	nc	nc	nc	nc

5. Conclusions

In this paper, a new family of iterative methods was designed to solve nonlinear equations from Ostrowski scheme, adding a Newton step with a "frozen" derivative and using a divided difference operator. This family, named CMT (α, β, γ), has a three-step iterative expression and three arbitrary parameters which can take any real or complex value.

In the convergence analysis of the new family, we obtained an order of convergence of four just like the order of the Ostrowski method. However, we managed to speed-up the convergence to six by setting the parameters β and γ as a function of α, resulting in an uniparametric CMT (α) family.

In the dynamical study, we constructed parameters spaces of the free critical points of the rational operator associated with the uniparametric family. These parameter spaces allowed us to understand the performance of the different members of the family, helping us to choose stable (for $\alpha = -1$, 0, 1, ...) and unstable (for $\alpha = -300$, 200, 400, ...) methods. Furthermore, we generated dynamical planes to show the behavior of these particular methods.

From numerical results, the order of convergence is verified by the ACOC, which is close to 6. The CMT (α) family proved to have an excellent numerical performance considering stable members as representatives. In general, this family has low errors and number of iterations to converge to the solution. However, the processing time (tcpu) varies depending on the nonlinear test functions used and the inherent complexity that the iterative schemes of the methods present when they are applied to said functions. In several cases, the tcpu of stable methods is significantly lower than other sixth-order methods developed so far. Furthermore, the methods for $\alpha = -1$, 0, 1 proved to be stable, have the lowest tcpu, and always converge to the solution for any initial estimate and nonlinear test function used. The methods for $\alpha = -300$, 200, 400 proved to be unstable, chaotic, have the highest tcpu, and tend not to converge to the solution according to the initial estimate and the nonlinear test function used. This verifies the theoretical results obtained in convergence analysis and dynamical study of CMT(α) family.

References

1. Neta, B. *Numerical Methods for the Solution of Equations*; Net-A-Sof: Monterey, CA, USA, 1983.
2. Petković, M.; Neta, B.; Petković, L.; Džunić, J. *Multipoint Methods for Solving Nonlinear Equations*, 1st ed.; Academic Press: Boston, MA, USA, 2013.
3. Amat, S.; Busquier, S. *Advances in Iterative Methods for Nonlinear Equations*; Springer: Cham, Switzerland, 2017.
4. Artidiello, S.; Cordero, A.; Torregrosa, J.; Penkova, M. Design and multidimensional extension of iterative methods for solving nonlinear problems. *Appl. Math. Comput.* **2017**, *293*, 194–203. [CrossRef]
5. Hunt, B.R.; Ott, E. Defining chaos. *Chaos Interdiscip. J. Nonlinear Sci.* **2015**, *25*. [CrossRef] [PubMed]
6. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
7. Scott, M.; Neta, B.; Chun, C. Basin attractors for various methods. *Appl. Math. Comput.* **2011**, *218*, 2584–2599. [CrossRef]
8. Amat, S.; Busquier, S.; Plaza, S. Review of some iterative root—Finding methods from a dynamical point of view. *SCIENTIA Ser. A Math. Sci.* **2004**, *10*, 3–35.
9. Blanchard, P. Complex analytic dynamics on the Riemann sphere. *Bull. Am. Math. Soc.* **1984**, *11*, 85–141. [CrossRef]
10. Alzahrani, A.; Behl, R.; Alshomrani, A. Some higher-order iteration functions for solving nonlinear models. *Appl. Math. Comput.* **2018**, *334*, 80–93. [CrossRef]
11. Chun, C.; Ham, Y. Some sixth-order variants of Ostrowski root-finding methods. *Appl. Math. Comput.* **2007**, *193*, 389–394. [CrossRef]
12. Amat, S.; Hernández, M.; Romero, N. Semilocal convergence of a sixth order iterative method for quadratic equations. *Appl. Numer. Math.* **2012**, *62*, 833–841. [CrossRef]

13. Noeiaghdam, S.; Sidorov, D.; Zamyshlyaeva, A.; Tynda, A.; Dreglea, A. A valid dynamical control on the reverse osmosis system using the CESTAC method. *Mathematics* **2021**, *9*, 48. [CrossRef]
14. Cordero, A.; Torregrosa, J.R. Variants of Newton's Method using fifth-order quadrature formulas. *Appl. Math. Comput.* **2007**, *190*, 686–698. [CrossRef]