mathematics

MDPI

Article

Memorizing Schröder's Method as an Efficient Strategy for Estimating Roots of Unknown Multiplicity

Alicia Cordero [1,*,†], Beny Neta [2,†] and Juan R. Torregrosa [1,†]

[1] Institute for Multidisciplinary Mathematics, Universitat Politècnica de València, 46022 València, Spain; jrtorre@mat.upv.es

[2] Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, USA; byneta@gmail.com

* Correspondence: acordero@mat.upv.es

† These authors contributed equally to this work.

Abstract: In this paper, we propose, to the best of our knowledge, the first iterative scheme with memory for finding roots whose multiplicity is unknown existing in the literature. It improves the efficiency of a similar procedure without memory due to Schröder and can be considered as a seed to generate higher order methods with similar characteristics. Once its order of convergence is studied, its stability is analyzed showing its good properties, and it is compared numerically in terms of their basins of attraction with similar schemes without memory for finding multiple roots.

Keywords: nonlinear equations; iterative methods with memory; multiple roots; derivative-free; efficiency; stability

check for updates

Citation: Cordero, A.; Neta, B.; Torregrosa, J.R. Memorizing Schröder's Method as an Efficient Strategy for Estimating Roots of Unknown Multiplicity. *Mathematics* **2021**, *9*, 2570. https://doi.org/10.3390/math9202570

Academic Editor: Luca Gemignani

Received: 24 September 2021
Accepted: 08 October 2021
Published: 13 October 2021

1. Introduction

There exist in the literature (see, for example, Reference [1–8]) numerous iterative methods without memory, involving or not derivatives, designed to estimate the multiple roots of a nonlinear equation $f(x) = 0$, but most of them need the knowledge of the multiplicity m of these roots.

It is well-known that Schröder method [9]:

$$x_{k+1} = x_k - \frac{f(x_k)f'(x_k)}{f'(x_k)^2 - f(x_k)f''(x_k)}, \; k = 0, 1, \ldots$$

is able to converge quadratically to the multiple solution of a nonlinear equation, that is, a value $\alpha \in \mathbb{R}$ such that $f(\alpha) = 0$ and $f^{(j)}(\alpha) = 0$, $j = 1, 2, \ldots, m-1$, with m being the multiplicity of the root. This scheme was originally deduced from Newton's scheme applied on the quotient $g(x) = \dfrac{f(x)}{f'(x)}$, and it is denoted along this manuscript by SM1,

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}, \; k = 0, 1, \ldots$$

Notice that SM1 requires 3 function evaluations per step. Similarly, the derivative-free Traub-Steffensen's method for g ,

$$x_{k+1} = x_k - \frac{g(x_k)}{g[x_k, x_k + \gamma g(x_k)]}, \; k = 0, 1, \ldots$$

with γ being a real parameter, requires 4 function evaluations per step and is no longer derivative-free. This Traub-Steffensen method on g is too expensive and is not further considered.

The main advantage of Schröder scheme is its independence of the knowledge of the multiplicity of the nonlinear function, in contrast with the modified Newton's method for multiple roots,

$$x_{k+1} = x_k - m\frac{f(x_k)}{f'(x_k)},\ k = 0, 1, \dots,$$

where m is the multiplicity of α, that must be known in this case. This scheme was also due to Schröder (also see Reference [9]), and we denote it by SM2. This scheme is second-order convergent and, therefore, optimal, in the sense of Kung-Traub conjecture, (as it uses two new functional evaluations per iteration; see Reference [10]). However, it needs the knowledge of the multiplicity, while SM1 does not use it; nevertheless, the main drawback of SM1 scheme is its low efficiency, as it needs to evaluate three nonlinear functions ($f(x)$, $f'(x)$ and $f''(x)$) per iteration.

Our aim in this manuscript is double: from one side, we would like to increase the efficiency of SM1 scheme, holding its ability for finding multiple roots of multiplicity m without knowing m and, from other side, to combine in the same algorithm the capability to find multiple roots with the use of more than one previous iterate. So, we propose an iterative scheme with memory for estimating multiple roots of unknown multiplicity. As far as we know, there is in the literature no iterative procedure satisfying these properties.

In the analysis of the convergence of the proposed scheme, some aspects must be taken into account, as it is an iterative method with memory, so that the error in several previous iterations must be considered and the multiplicity of the root m should also be a key element of the demonstration, although its specific value is not known. Regarding this fact, it should be noticed that $f^{(q)}(\alpha) = 0$ for $q = 1, 2, \dots, m-1$ and $f^{(m)}(\alpha) \neq 0$. So, the Taylor expansions around α of f and f' appearing in the iterative expression should take this information into account.

On the other hand, as our proposed scheme is an iterative procedure that uses three previous iterates for calculating the next one, it is necessary to express the error equation in terms of their corresponding errors and, from it, to deduce its order of convergence. This is made by means of a classical result by Ortega and Rheinboldt [11], that is presented below.

Theorem 1. *Let ψ be an iterative method with memory that generates a sequence $\{x_k\}$ of approximations to the root α, and let this sequence converges to α. If there exist a nonzero constant η and positive numbers t_i, $i = 0, 1, \dots, m$, such that the inequality*

$$|e_{k+1}| \leq \eta \prod_{i=0}^{m} |e_{k-i}|^{t_i}$$

holds, then the R-order of convergence of the iterative method ψ satisfies the inequality

$$O_R(\psi, \alpha) \geq p,$$

where p is the unique positive root of the equation

$$p^{m+1} - \sum_{i=0}^{m} t_i p^{m-i} = 0.$$

In this manuscript, Section 2 is devoted to the design and convergence analysis of the proposed derivative-free iterative method with memory to find multiple roots (without the knowledge of its multiplicity). In Section 3, its stability is analyzed in order to deduce its dependence on the initial estimations for both simple and multiple roots. In Section 4, the numerical performance of the method is checked on several test functions, being analyzed, as well as their corresponding basins of attraction, in comparison with existing Schröder methods.

2. Design and Convergence Analysis

Our starting point is the derivative-free scheme with memory due to Traub [12],

$$x_{k+1} = x_k - \frac{f(x_k)}{f[x_{k-2}, x_k] - f[x_{k-2}, x_{k-1}] + f[x_{k-1}, x_k]}, \; k = 0, 1, \ldots, \tag{1}$$

with x_0, x_{-1} and x_{-2} being their initial estimations, with order of convergence $p = 1.839$. To estimate the multiple roots of $f(x) = 0$, we define the auxiliary function $g(x) = \dfrac{f(x)}{f'(x)}$, and then we apply Traub's method (1), on $g(x) = 0$, getting what we call gTM method,

$$x_{k+1} = x_k - \frac{g(x_k)}{g[x_{k-2}, x_k] - g[x_{k-2}, x_{k-1}] + g[x_{k-1}, x_k]}, \; k = 0, 1, \ldots, \tag{2}$$

an iterative scheme with memory that is proven to converge to any multiple roots of f with the same order than original Traub's scheme, without the knowledge of its multiplicity m and using two new functional evaluations per iteration.

Theorem 2. *Let $f : \mathbb{C} \to \mathbb{C}$ be an analytic function in the neighborhood of the multiple zero α of f with unknown multiplicity $m \in \mathbb{N} - \{1\}$. Then, for an initial guess x_0 sufficiently close to α, the iteration function gTM defined in (2) has order of convergence $\frac{1}{3}\left(\sqrt[3]{3\sqrt{33}+19} + \sqrt[3]{19 - 3\sqrt{33}} + 1\right) \approx 1.83929$, with its error equation being*

$$e_{k+1} = -\frac{(m+1)c_1^2 - 2mc_2}{m^2} e_{k-2}e_{k-1}e_k + O_3(e_{k-2}, e_{k-1}, e_k),$$

where $c_j = \frac{m!}{(m+j)!}\frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}$, $j = 1, 2, 3, \ldots$, and $O_3(e_{k-2}, e_{k-1}, e_k)$ denotes the terms of the error equation with products of powers of e_{k-2}, e_{k-1} and e_k, whose exponents sum at least 3.

Proof. Let α be a multiple zero of $f(x)$, and $e_k = x_k - \alpha$ be the error at kth-iterate. Expanding $f(x_k)$ and $f'(x_k)$ of $x = \alpha$ by using the Taylor's series expansion, we have

$$f(x_k) = \frac{f^{(m)}(\alpha)}{m!} e_k^m \left[1 + c_1 e_k + c_2 e_k^2 + O(e_k^3)\right], \tag{3}$$

and

$$f'(x_k) = \frac{f^{(m)}(\alpha)}{m!} e_k^{m-1} \left[1 + (m+1)c_1 e_k + (m+2)c_2 e_k^2 + O(e_n^3)\right], \tag{4}$$

where $c_j = \frac{m!}{(m+j)!}\frac{f^{(m+j)}(\alpha)}{f^{(m)}(\alpha)}$, $j = 1, 2, 3, \ldots$

By using expressions (3) and (4), we have

$$g(x_k) = \frac{f(x_k)}{f'(x_k)} = \frac{1}{m}\left(e_k - \frac{1}{m}c_1 e_k^2 + \frac{(m+1)c_1^2 - 2mc_2}{m^2} e_k^3\right) + O(e_k^4).$$

In a similar way,

$$g(x_{k-1}) = \frac{1}{m}\left(e_{k-1} - \frac{1}{m}c_1 e_{k-1}^2 + \frac{(m+1)c_1^2 - 2mc_2}{m^2} e_{k-1}^3\right) + O(e_{k-1}^3),$$

and

$$g(x_{k-2}) = \frac{1}{m}\left(e_{k-2} - \frac{1}{m}c_1 e_{k-2}^2 + \frac{(m+1)c_1^2 - 2mc_2}{m^2} e_{k-2}^3\right) + O(e_{k-2}^3).$$

Then,

$$\begin{aligned} A &= \frac{g(x_{k-2}) - g(x_k)}{x_{k-2} - x_k} \\ &= \frac{1}{m}\left(1 - \frac{1}{m}c_1(e_{k-2} + e_k) + \frac{(m+1)c_1^2 - 2mc_2}{m^2}\left(e_{k-2}^2 + e_{k-2}e_k + e_k^2\right)\right) + O_3(e_{k-2}, e_k), \end{aligned}$$

where $O_3(e_{k-2}, e_k)$ denotes that the neglected terms of the error equation have products of powers of e_{k-2} and e_k, whose exponents sum at least 3.

$$\begin{aligned} B &= \frac{g(x_{k-2}) - g(x_{k-1})}{x_{k-2} - x_{k-1}} \\ &= \frac{1}{m}\left(1 - \frac{1}{m}c_1(e_{k-2} + e_{k-1}) + \frac{(m+1)c_1^2 - 2mc_2}{m^2}\left(e_{k-2}^2 + e_{k-2}e_{k-1} + e_{k-1}^2\right)\right) + O_3(e_{k-2}, e_{k-1}), \end{aligned}$$

and

$$\begin{aligned} C &= \frac{g(x_{k-1}) - g(x_k)}{x_{k-1} - x_k} \\ &= \frac{1}{m}\left(1 - \frac{1}{m}c_1(e_{k-1} + e_k) + \frac{(m+1)c_1^2 - 2mc_2}{m^2}\left(e_{k-1}^2 + e_{k-1}e_k + e_k^2\right)\right) + O_3(e_{k-1}, e_k). \end{aligned}$$

Therefore, from expression (2),

$$\begin{aligned} e_{k+1} &= e_k - \frac{g(x_k)}{A - B + C} = \frac{\left(e_k - \frac{1}{m}c_1 e_k^2 + \frac{(m+1)c_1^2 - 2mc_2}{m^2}e_k^3\right) + O(e_k^4)}{1 - \frac{1}{m}2c_1 e_k + \frac{(m+1)c_1^2 - 2mc_2}{m^2}\left(e_{k-2}e_k - e_{k-2}e_{k-1} + e_{k-1}e_k + 2e_k^2\right)} \\ &= -\frac{(m+1)c_1^2 - 2mc_2}{m^2}e_{k-2}e_{k-1}e_k + O_3(e_{k-2}, e_{k-1}, e_k), \end{aligned}$$

and then the order of convergence is the only real root of polynomial $p^3 - p^2 - p - 1$, that is, $\frac{1}{3}\left(\sqrt[3]{3\sqrt{33} + 19} + \sqrt[3]{19 - 3\sqrt{33}} + 1\right) \approx 1.83929$, by applying Theorem 1. So, the proof is finished. □

The main advantage of this scheme is its ability for finding simple, as well as multiple, roots of a nonlinear function without the knowledge of the multiplicity, with better efficiency as SM1. Certainly, by using Ostrowski's efficiency index [13], $I_{SM1} = 2^{\frac{1}{3}} \approx 1.25992$ is lower than $I_{gTM} = 1.84^{\frac{1}{2}} \approx 1.35647$, where each index I is calculated as $p^{\frac{1}{d}}$, with p being the order of convergence of the method, and d the amount of new functional evaluations per iteration.

In the next section, a dynamical analysis is made on this scheme, in order to show its qualitative performance on simple and multiple roots. As it is an iterative method with memory, multidimensional real dynamics must be used.

3. Qualitative Study of the Proposed Iterative Methods with Memory for Multiple Roots

Let us remark that our method uses three previous iterations in order to generate the following one; therefore, it can be expressed in general as

$$x_{k+1} = \Upsilon(x_{k-2}, x_{k-1}, x_k), \;\; k \geq 0,$$

where x_0, x_{-1} and x_{-2} are the initial estimations. By means of the procedure defined in Reference [14], this method can be described as a discrete real multidimensional dynamical system, and its qualitative behavior can be analyzed.

The qualitative performance of the dynamical system has a key element in the characterization of their fixed points, in terms of stability. In order to calculate the fixed points of Υ, an auxiliary vectorial function $M : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ can be defined, related to Υ by means of:

$$M(x_{k-2}, x_{k-1}, x_k) = (x_{k-1}, x_k, \Upsilon(x_{k-2}, x_{k-1}, x_k)), \;\; k = 0, 1, 2, \ldots.$$

Therefore, a fixed point of M is obtained if not only $x_{k+1} = x_k$, but also $x_{k-1} = x_k$ and $x_{k-2} = x_{k-1}$.

Fixed points (w, z, x) of M satisfy $w = z = x$ and $x = \Upsilon(w, z, x)$. So, $w = x_{k-2}$, $z = x_{k-1}$ and $x = x_k$. In the following, we define some basic dynamical concepts as direct extension of those used in complex discrete dynamics analysis (see Reference [15]).

Let us consider $M : \mathbb{R}^3 \to \mathbb{R}^3$ a vectorial rational function obtained by the application of an iterative method on a scalar polynomial $p(x)$. Then, if a fixed point (w, z, x) of operator M is different from (r, r, r), where r is a zero of $p(x)$, then the fixed point is called strange. Moreover, the orbit of a point $x^* \in \mathbb{R}^3$ is the set of successive images from x^* by the vector function, that is, $\mathcal{O}(x^*) = \{x^*, M(x^*), \ldots, M^n(x^*), \ldots\}$. Indeed, a point $x^* \in \mathbb{R}^3$ is called periodic with period p if $M^p(x^*) = x^*$ and $M^q(x^*) \neq x^*$, for $q = 1, 2, \ldots, p-1$. We should notice that a fixed point is a 1-periodic point.

It is also known that the qualitative performance of a fixed point of M is classified in terms of its asymptotical behavior. It can be analyzed by means of the Jacobian matrix M', as is stated in the next result (see, for instance, Reference [16]).

Theorem 3. *Let M from $\mathbb{R}^m$ to $\mathbb{R}^m$ be of class $\mathcal{C}^2$. Let us also assume that x^* is a k-periodic point. Let $\lambda_1, \lambda_2, \ldots, \lambda_m$ be the eigenvalues of the Jacobian matrix $M'(x^*)$ at periodic point x^*. Then, it holds that:*

(a) *If all the eigenvalues λ_j verify $|\lambda_j| < 1$, then x^* is attracting.*

(b) *If one eigenvalue λ_{j_0} verifies $|\lambda_{j_0}| > 1$, then x^* is unstable, that is, repelling or saddle.*

(c) *If all the eigenvalues λ_j verify $|\lambda_j| > 1$, then x^* is repelling.*

Moreover, if there exist an eigenvalue λ_i of the Jacobian matrix M' evaluated at a fixed point x^* satisfying $|\lambda_i| < 1$ and another one λ_j such that $|\lambda_j| > 1$, then, x^* is called saddle fixed point. As an extension of the concept in one-dimensional dynamics, if the eigenvalues of $M'(x^*)$ satisfy $|\lambda_j| = 0$ for all values of $j = 1, 2, \ldots, m$, then, the fixed point x^* is not only attracting but also superattracting. Therefore, the method has quadratic convergence, at least on the class of nonlinear functions that derive the rational function (see Reference [12]).

By considering x^* an attracting fixed point of M, its basin of attraction $\mathcal{A}(x^*)$ is defined as the set of preimages of any order

$$\mathcal{A}(x^*) = \Big\{ x_0 \in \mathbb{R}^3 : M^m(x_0) = x^*, \text{ for some } m \in \mathbb{N} \Big\}.$$

The qualitative performance of different iterative schemes designed for solving nonlinear equations with multiple roots has been studied by different authors (see, for example, Reference [17–19]). It has been made by using discrete complex dynamics, as all these schemes are without memory. In these studies, it has been obtained that, when an iterative method (without memory) designed for finding multiple roots acts on a nonlinear function with both simple and multiple roots, it is quite usual that the basins of attraction of simple roots are narrower than those of multiple roots. Indeed, it may happen that those simple roots define fixed points of the rational function that are repulsive. Therefore, the iterative method should be able to find only multiple roots.

The following qualitative analysis is made on $p(x) = (x+1)(x-1)^m$, $m \geq 1$, so that the capability of the scheme to find both simple and multiple roots (with multiplicity m) is tested.

Theorem 4. *The multidimensional rational operator associated with method gTM, when it is applied on polynomial $p(x) = (x+1)(x-1)^m$, is*

$$TM(w,z,x) = \left(z, x, \frac{m^2(x+1)(z+1)(w+1) + 2m\left(-2x^2 + x(z+w) + zw - 1\right) - (x-1)(z-1)(w-1)}{m^2(x+1)(z+1)(w+1) + 2m(x(zw-3) + z + w) + (x-1)(z-1)(w-1)}\right).$$

Indeed, the rational operator TM satisfies that there are one strange fixed point with equal components to $\frac{1-m}{1+m}$, that is saddle. Moreover, both fixed points corresponding to the roots of $p(x)$ are superattracting.

Proof. By definition of the multidimensional dynamical system,

$$TM(w,z,x) = (z, x, gTM(w,z,x)),$$

with $gTM(w,z,x) = x + \frac{g(x)}{g[w,x]-g[w,z]+g[z,x]}$, where $g(x) = \dfrac{p(x)}{p'(x)}$. Therefore,

$$TM(w,z,x) = \left(z, x, \frac{m^2(x+1)(z+1)(w+1) + 2m\left(-2x^2 + x(z+w) + zw - 1\right) - (x-1)(z-1)(w-1)}{m^2(x+1)(z+1)(w+1) + 2m(x(zw-3) + z + w) + (x-1)(z-1)(w-1)}\right)$$

is obtained. In order to calculate the fixed points of $TM(w,z,x)$, equation $TM(w,z,x) = (w,z,x)$ must be solved. By means of algebraic manipulations, it is reduced to $w = z = x$ and

$$\frac{(-1+m+x+mx)(-1+x^2)}{(-1+x)^2 + m(1+x)^2} = 0.$$

So, the only fixed points are the roots of $p(x)$ and the strange fixed point $w = z = x = \frac{1-m}{1+m}$ that depends on the multiplicity of the root. In order to analyze the stability of the fixed points, we calculate the Jacobian matrix

$$TM'(w,z,x) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ R_1 & R_2 & R_3 \end{pmatrix},$$

where

$$R_1 = \frac{4m(m+1)\left(x^2-1\right)(mz+m+z-1)(x-z)}{\left(m^2(x+1)(z+1)(w+1) + 2m(x(zw-3)+z+w) + (x-1)(z-1)(w-1)\right)^2},$$

$$R_2 = \frac{4m(m+1)\left(x^2-1\right)(mw+m+w-1)(x-w)}{\left(m^2(x+1)(z+1)(w+1) + 2m(x(zw-3)+z+w) + (x-1)(z-1)(w-1)\right)^2},$$

and

$$R_3 = -\frac{4m\left(m^2(z+1)(w+1)r(x,z,w) + 2mq(x,z,w) + (z-1)(w-1)s(x,z,w)\right)}{\left(m^2(x+1)(z+1)(w+1) + 2m(x(zw-3)+z+w) + (x-1)(z-1)(w-1)\right)^2},$$

where $r(x,z,w) = (x(x+2) + z(w-1) - w - 2)$, $q(x,z,w) = x^2(zw-3) + 2x(z+w) + \left(z^2-1\right)w^2 - z^2 - 3zw + 2$, and $s(x,z,w) = ((x-2)x + zw + z + w - 2)$.

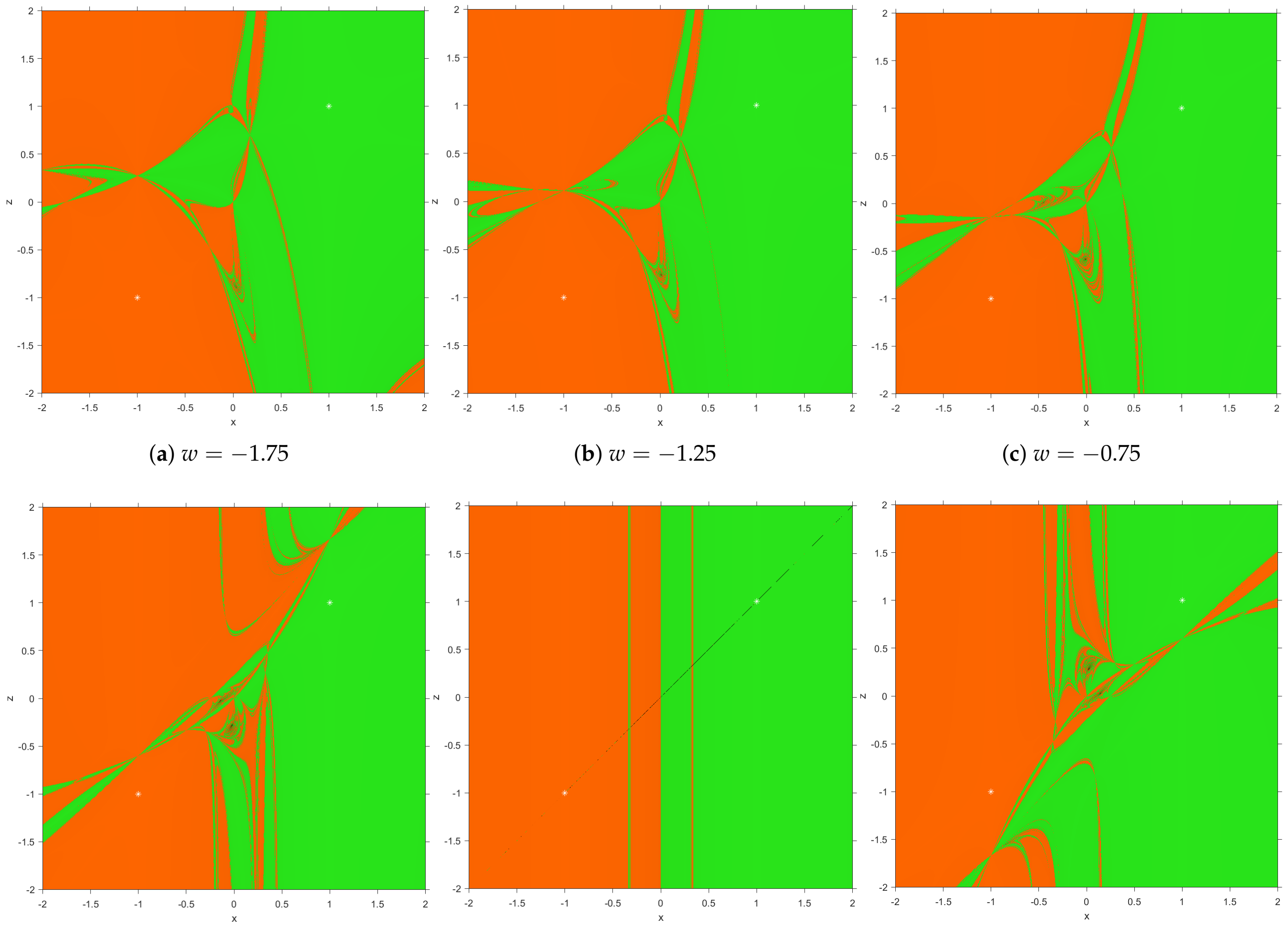
The eigenvalues of $TM'(w,z,x)$, when $(w,z,x) = (1,1,1)$ or $(w,z,x) = (-1,-1,-1)$, are all equal to zero. Then, these are superattracting fixed points.

Regarding the strange fixed point $(\frac{1-m}{1+m}, \frac{1-m}{1+m}, \frac{1-m}{1+m})$, in order to avoid an indetermination, it is necessary to calculate a simplified rational operator, forcing to $w = x = z$;

therefore, the reduced Jacobian matrix has two zero eigenvalues, and the third one is $2 > 0$. So, the strange fixed point is always a saddle point and, therefore, is in the boundary of the basins of attraction. $\square$

A very useful tool to visualize the analytical results is the dynamical plane of the system, composed by the set of the different basins of attraction. Here, the dynamical plane of the proposed method gTM is built by calculating the orbit of a mesh of 800×800 starting points (z, x) for a fixed value of w in the starting grid. As the iterative schemes needs to be started with three initial estimations, we generate a mesh of dynamical planes, each one of them with a fixed value of w in the interval $[-1.75, 1.75]$. In these phase portraits, each point of the mesh is painted in different colors (orange and green in this case), depending on the attractor they converge to (marked as a white star), with a tolerance of 10^{-3}. In addition, they appear in black color if the orbit has not reached any attracting fixed point in a maximum of 500 iterations. As the fixed value of w is changed in a vector of values belonging to $[-1.75, 1.75]$, it yields to a composition of figures for each multiplicity, giving rise to a kind of contour plots.

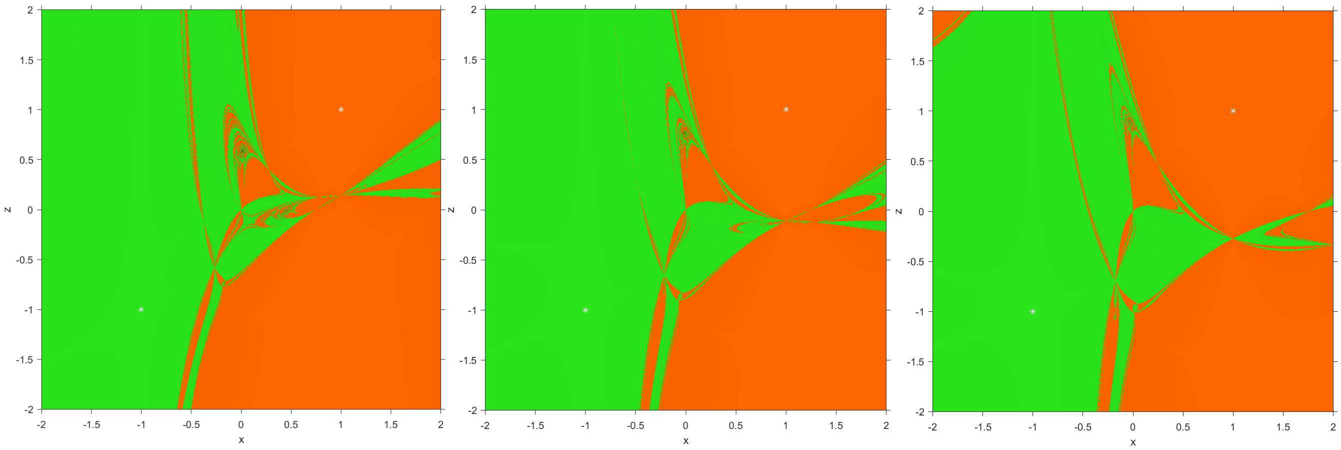
In Figure 1, we show the performance of gTM scheme on $p(x)$, that is, of rational operator TM for simple roots. Observing the behavior for the different plots with the three first iterations varying each in $[-2, 2]$, the stable feasibility is noticed. The basins of attraction of the roots are the only ones; they are wide, and the only different performance (better than others in terms of the simplicity of the boundary among the basins) is the case $w = 0$, where the rational function is simplified. In all cases, it is observed that the only possible behavior of method gTM is the convergence to the roots.



(**a**) $w = -1.75$ (**b**) $w = -1.25$ (**c**) $w = -0.75$

(**d**) $w = -0.25$ (**e**) $w = 0$ (**f**) $w = 0.25$

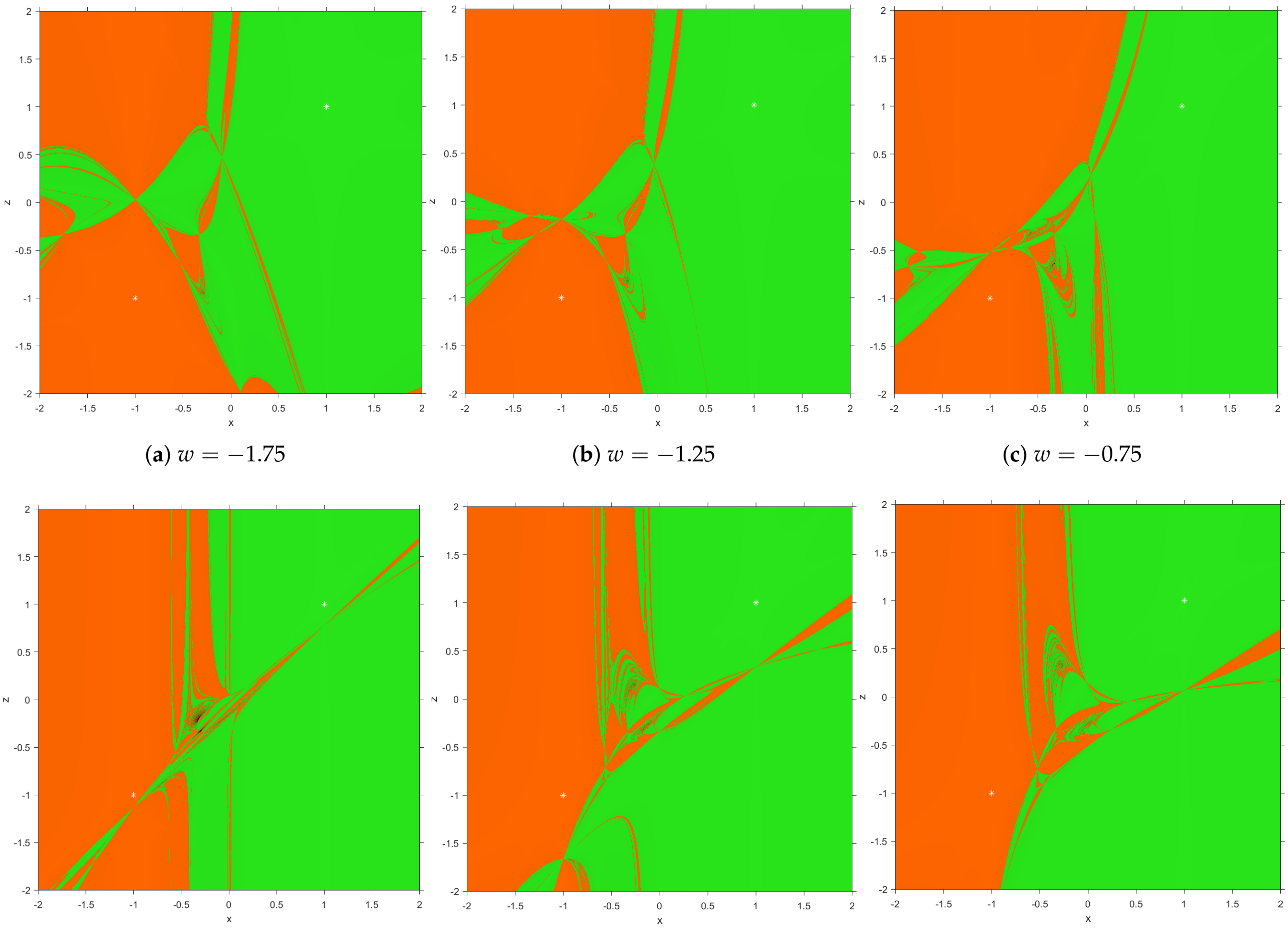
Figure 1. *Cont.*



(**g**) $w = 0.75$ (**h**) $w = 1.25$ (**i**) $w = 1.75$

Figure 1. Dynamical planes of TM rational operator on $p(x)$, for $m = 1$.

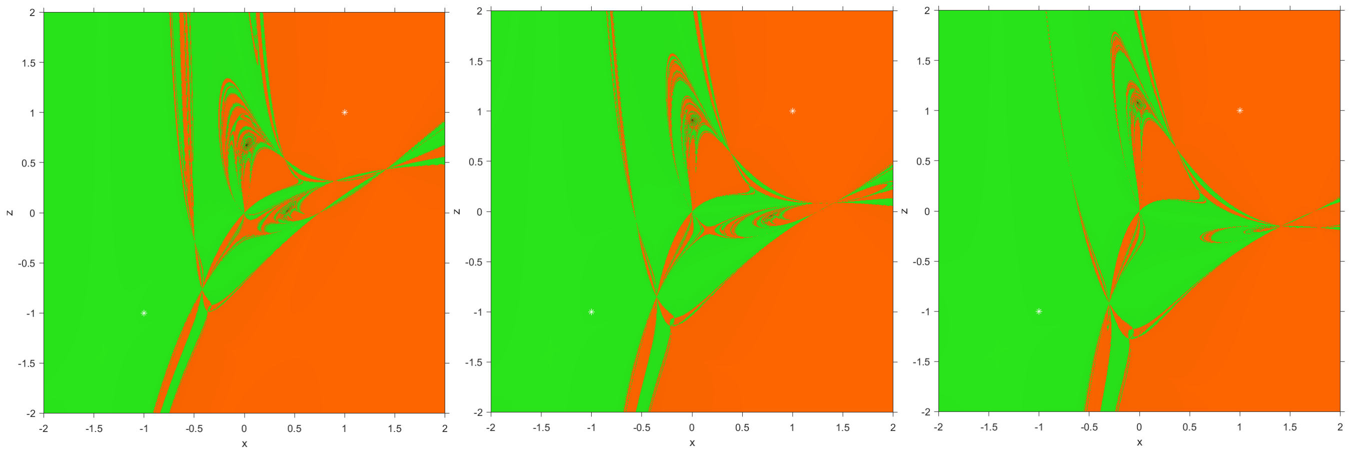
On the other hand, in Figure 2, we show a very similar performance when one of the roots is double, and the other one is simple. The basins of attraction are equally wide, and this behavior is very similar when other multiplicities have been explored. In addition, in this case can be seen that there is only convergence to the roots, as darker areas are only slower convergence, due to the higher complexity of the boundary of the basins of attraction.



(**a**) $w = -1.75$ (**b**) $w = -1.25$ (**c**) $w = -0.75$

(**d**) $w = -0.25$ (**e**) $w = 0$ (**f**) $w = 0.25$

Figure 2. *Cont.*



(**g**) $w = 0.75$ (**h**) $w = 1.25$ (**i**) $w = 1.75$

Figure 2. Dynamical planes of TM rational operator on $p(x)$, for $m = 2$.

In the next section, the numerical and dynamical performance of our proposed scheme is tested on several nonlinear functions of increasing complexity.

4. Numerical Performance and Dynamical Tests

In this section, we compare three methods, namely SM2 (requiring the knowledge of the multiplicity), SM1, and gTM (derived from Traub's method). The last two methods do not require the knowledge of the multiplicity, but they do require extra functional evaluations per iteration step (three in case of SM1, two in gTM case).

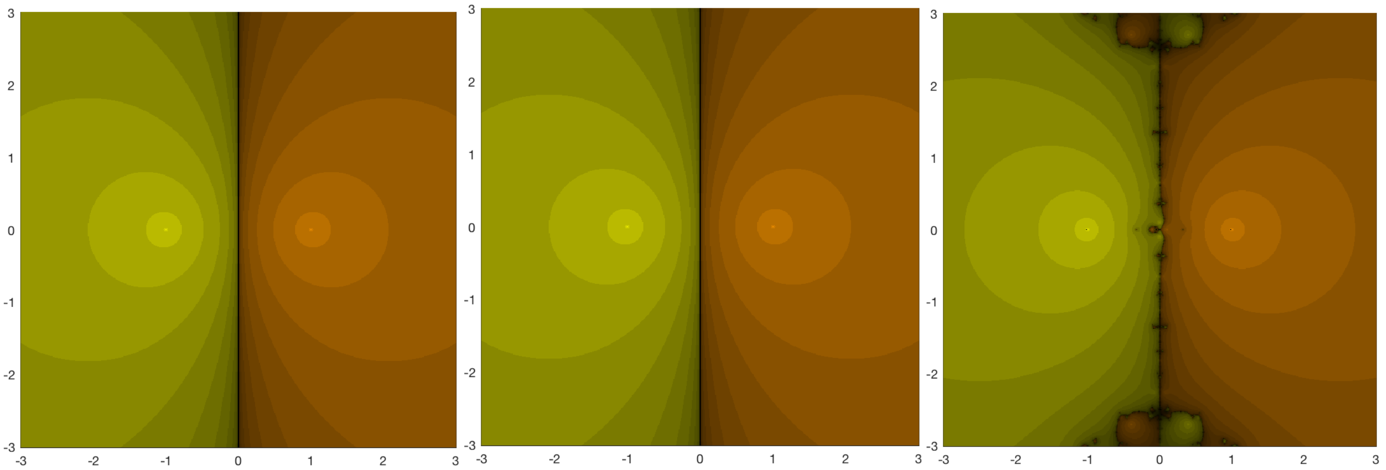
The methods are compared both qualitatively via the basins of attraction figures and quantitatively via several measures. These measures are: the CPU run-time to run the method on points in a 6 by 6 square centered in the origin. We divided the square by uniformly distributed horizontal and vertical lines and took all points of intersection as initial points for the iterative process. For gTM, a method with memory, we had to take two additional starting points $x_{-1} = x_0 + d$ and $x_{-2} = x_0 + 2d$, where d is the spacing of the lines. Another criterion collected by the code is the average number of iterations per point (AIPP), but, since the methods require different number of functional evaluations per step, we took the average number of functions per point (AFPP). The third criterion is the number of divergent points (DP), which is the number of points for which the method did not converge in 40 iterations using a tolerance of 10^{-7}.

The functions used for our comparative study are:

- $f_1(x) = (z^2 - 1)^3$,
- $f_2(x) = (z^3 - 1)^4$,
- $f_3(x) = (z^4 - 1)^2$,
- $f_4(x) = (z^5 - 1)^3$,
- $f_5(x) = (z - i)^3(e^{z+i} - 1)^3$,
- $f_6(x) = (z^7 - 1)^4$.

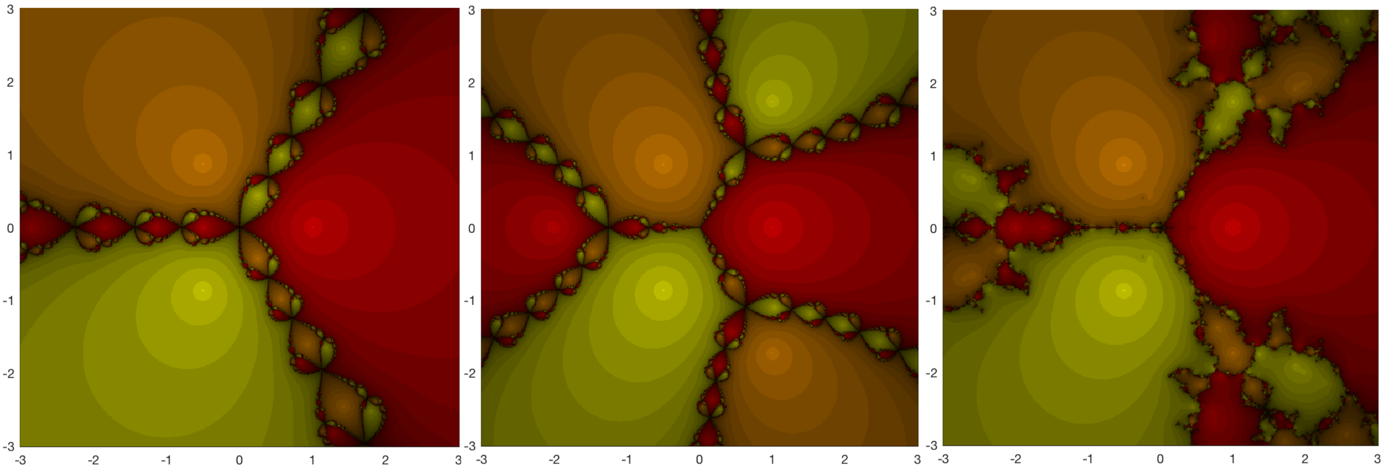
Notice that all but one are polynomials of various degrees and various multiplicities.

In Figures 3–8, we have plotted the basins of attraction for the 3 methods for each test function. Each figure has 3 sub-figures, with the left-most being the Schröder method using the multiplicity (SM2), the middle one Schröder method not requiring the knowledge of multiplicity (SM1), and the right-most Traub's method for multiple roots (gTM).



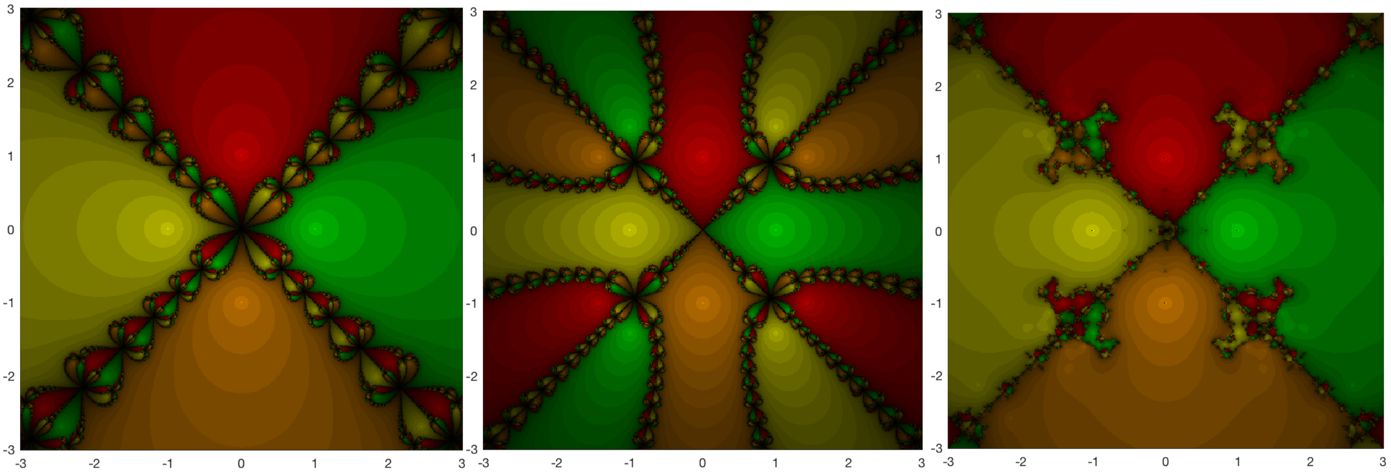
(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 3. Dynamical planes of Schröder methods and gTM on $f_1(x)$.



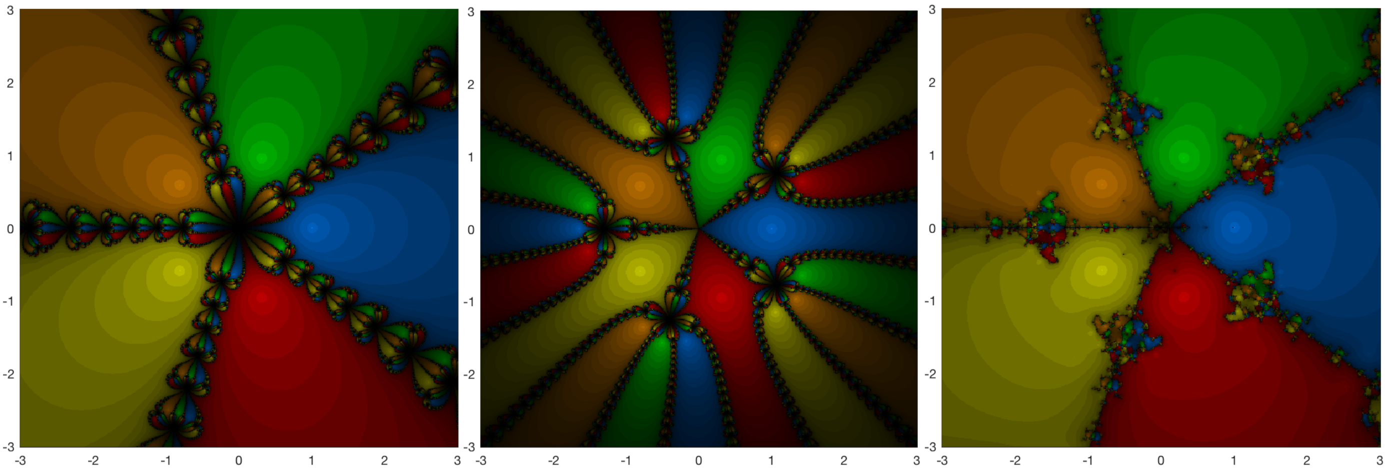
(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 4. Dynamical planes of Schröder methods and gTM on $f_2(x)$.



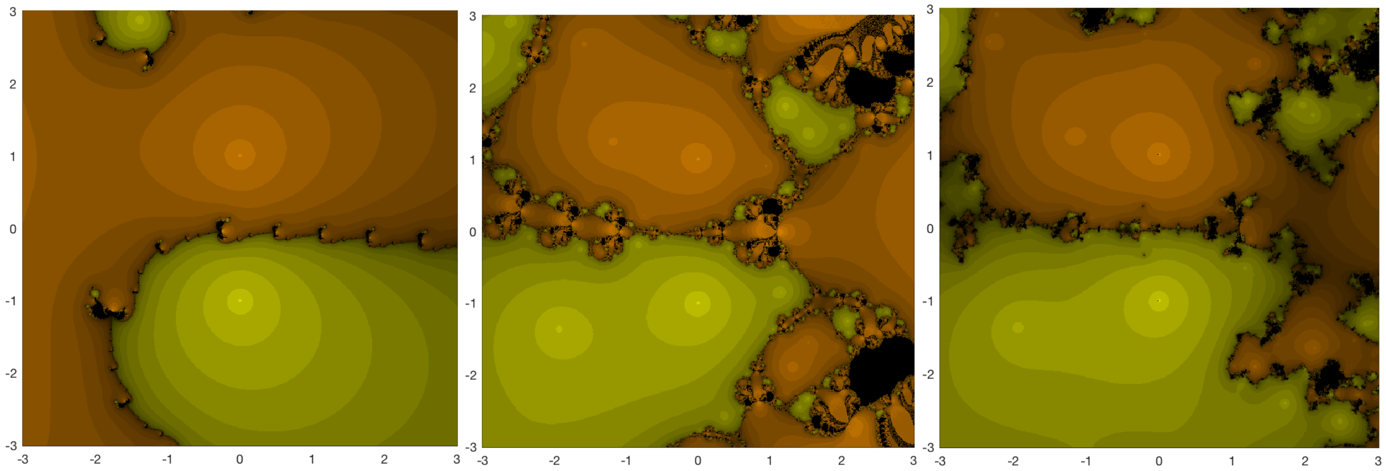
(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 5. Dynamical planes of Schröder methods and gTM on $f_3(x)$.



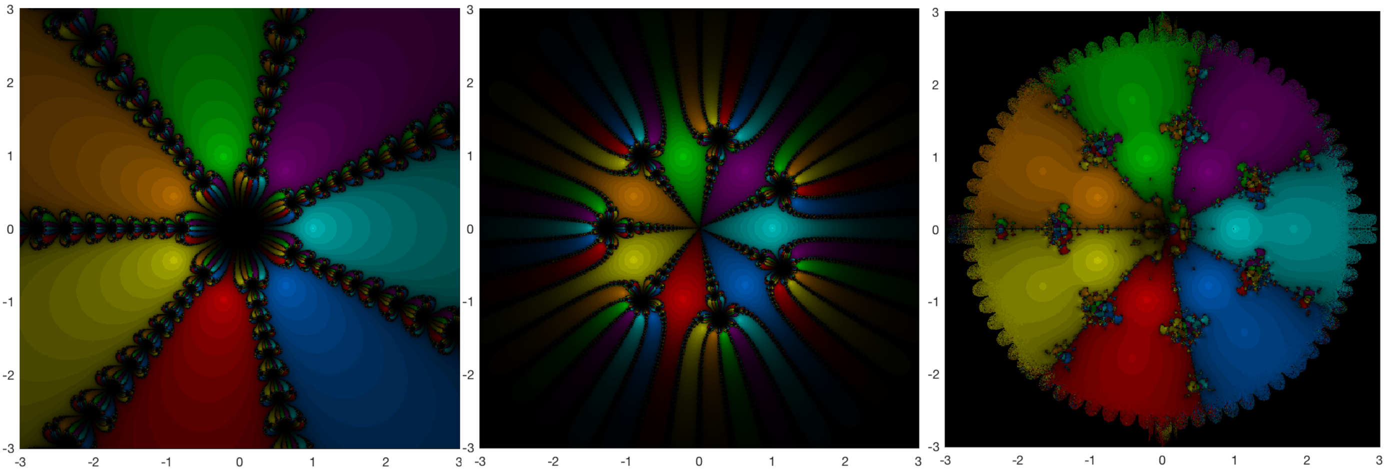
(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 6. Dynamical planes of Schröder methods and gTM on $f_4(x)$.



(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 7. Dynamical planes of Schröder methods and gTM on $f_5(x)$.



(**a**) SM2 (**b**) SM1 (**c**) gTM

Figure 8. Dynamical planes of Schröder methods and gTM on $f_6(x)$.

Based on Figure 3, it is clear that SM1 and SM2 have similar basins, and gTM has more lobes on the boundary between the two basins. From Figure 4, we notice that gTM is better than SM1. In the next 3 figures, gTM is best, with wider basins of attraction and narrower black areas of no convergence to the roots. This performance is held even for non-polynomial function f_5. Moreover, in Figure 8, it can be noticed that the basins of attraction of method SM2 are wider than our gTM method.

We now refer to the data in Tables 1–3. The CPU run-time in seconds is given in Table 2. It is clear that SM2 is consistently faster than the others. If the multiplicity is not known, then gTM is faster than SM1, except for the first example. On average, gTM is faster than SM1.

Table 1. CPU run-time (sec) for each method on the test functions.

Methods	f_1	f_2	f_3	f_4	f_5	f_6	**Average**
SM2	126.93	224.61	288.52	406.23	313.50	674.53	339.05
SM1	194.09	431.38	529.08	1013.37	632.83	2183.55	830.72
gTM	267.12	408.12	424.59	515.34	582.81	709.32	484.55

Table 2. Average number of function-evaluations per point (AFPP) for each method on the test functions.

Methods	f_1	f_2	f_3	f_4	f_5	f_6	**Average**
SM2	11.65	15.21	20.37	22.22	13.69	28.30	18.574
SM1	17.48	24.72	35.46	48.56	25.11	81.92	38.87
gTM	13.72	17.68	18.48	18.40	18.20	40.34	21.13

Table 3. Number of divergent points (DP) for each method on the test functions.

Methods	f_1	f_2	f_3	f_4	f_5	f_6	**Average**
SM2	601	8	2449	5158	1529	20299	5007.33
SM1	601	19	2529	8522	21253	79139	18677.17
gTM	9	20	41	241	11483	127078	23145.33

The average number of function evaluations per point (see Table 2) is the highest for SM1 for all examples. Note that the last example is the hardest for all methods. The number of divergent points is the lowest for gTM for examples 1, 3, and 4. SM1 has the most divergent points for the first 6 examples, but, on the last example, gTM performed poorly and became third place overall. The method SM2 was best, on average, for the 3 categories followed by gTM for 2 categories.

5. Conclusions

A new iterative scheme with memory with ability for finding both simple and multiple roots (without the need of knowing its multiplicity) have been constructed. It is, as far as we know, the first method with these properties in the literature. Its order of convergence have been proven to be approximately 1.84 with two new functional evaluations per iteration; this yields the scheme to improve the efficiency of the Schröder scheme without memory SM1, that has similar properties. Using multidimensional real discrete dynamics and low-degree polynomials with simple and multiple roots, the stability of the proposed scheme has been analyzed, showing wide areas of convergence to both kind of roots.

In the last section, Schröder and gTM methods running on several examples have allowed us to conclude that, if the multiplicity is known in advance, then, SM1 and gTM cannot compete, even though gTM is better than SM1. However, when the multiplicity is not known, proposed method gTM shows a very good performance and better efficiency than SM1 methods, in terms of execution time, computational cost, and wideness of the basins of attraction.

Author Contributions: Conceptualization, A.C. and J.R.T.; methodology, B.N.; software, A.C. and B.N.; validation, B.N.; formal analysis, J.R.T.; investigation, A.C. ; writing—original draft preparation, A.C. and B.N.; writing—review and editing, J.R.T. ; supervision, B.N. and J.R.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by PGC2018-095896-B-C22 (MCIU/AEI/FEDER, UE).

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Acknowledgments: The authors would like to thank the anonymous reviewers for their suggestions and comments that have improved the final version of this manuscript.

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

References

1. M. Petković, B. Neta, L. Petković, J. Džunić, Multipoint methods for solving nonlinear equations, Academic press,Oxford, UK, 2013.
2. Amat, S.; Busquier, S. *Advances in Iterative Methods for NONlinear Equations*; SEMA SIMAI Springer Series 10; Springer: Cham, Switzerland, 2016.
3. Behl, R.; Cordero, A.; Torregrosa, J.R. A new higher-order optimal derivative-free scheme for multiple roots. *J. Comput. Appl. Math.* **2021**, 113773, in press, doi:10.1016/j.cam.2021.113773.
4. Kumar, S.; Kumar, D.; Sharma, J.R.; Cesarano, C.; Aggarwal, P.; Chu, Y.M. An optimal fourth order derivative-free numerical algorithm for multiple roots. *Symmetry* **2020**, *12*, 1038, doi:10.3390/sym12061038.
5. Akram, S.; Akram, F.; Junjua, M.; Arshad, M.; Afzal, T. A family of optimal eighth order iterative function for multiple roots and its dynamics. *J. Math.* **2021**, *77*, 1249–1272.
6. Sharma, J.R.; Arora, H. A family of fifth-order iterative methods for finding multiple roots of nonlinear equations. *Numer. Anal. Appl.* **2021**, *14*, 186–199.
7. Kumar, S.; Kumar, D.; Sharma, J.R.; Argyros, I.K. An efficient class of fourth-order derivative-free method for multiple roots. *Int. J. Nonlinear Sci. Numer. Simul.* **2021**, doi:10.0505/ijnsns-2020-0161.
8. Zafar, F.; Cordero, A.; Torregrosa, J.R. A family of optimal fourth-order method for multiple roots of nonlinear equations. *Math. Methods Appl. Sci.* **2020**, *43*, 7869–7884.
9. Schröder, E. Über unendlich viele Algorithmen zur Auflösung der Gleichungen. *Math. Ann.* **1870**, *2*, 317–365.
10. Kung, H.T.; Traub, J.F. Optimal order of one-point and multipoint iteration. *J. Assoc. Comput. Mach.* **1974**, *21*, 643–651.
11. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: Cambridge, MA, USA, 1970.
12. Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice-Hall: Hoboken, NJ, USA, 1964.
13. Ostrowski, A.M. *Solutions of Equations and Systems of Equations*; Academic Press: New York, NY, USA; London, UK, 1966.
14. Campos, B.; Cordero, A.; Torregrosa, J.R.; Vindel, P. A multidimensional dynamical approach to iterative methods with memory. *Appl. Math. Comput.* **2015**, *271*, 701–715.
15. Devaney, R.L. *An Introduction to Chaotic Dynamical Systems*; Advances in Mathematics and Engineering; CRC Press: Boca Raton, FL, USA, 2003.
16. Robinson, R.C. *An Introduction to Dynamical Systems, Continous and Discrete*; American Mathematical Society: Providence, RI, USA, 2012.
17. Chicharro, F.I.; Contreras, R.A.; Garrido, N. A Family of Multiple-Root Finding Iterative Methods Based on Weight Functions. *Mathematics* **2020**, *8*, 2194, doi:10.3390/math8122194.
18. Neta, B. A New Derivative-Free Method to Solve Nonlinear Equations. *Mathematics* **2021**, *9*, 583, doi:10.3390/math9060583.
19. Geum, Y.H.; Kim, Y.I.; Neta, B. A sixth-order family of three-point modified Newton-like multiple-root finders and the dynamics behind their extraneous fixed points. *Appl. Math. Comput.* **2016**, *283*, 120–140.