Fixed Point Theory, 24(2023), No. 1, 155-172
DOI: 10.24193/fpt-ro.2023.1.07
http://www.math.ubbcluj.ro/~nodeacj/sfptcj.html

IMPROVING THE ORDER OF A FIFTH-ORDER FAMILY OF VECTORIAL FIXED POINT SCHEMES BY INTRODUCING MEMORY

A. CORDERO*, N. GARRIDO*, J.R. TORREGROSA* AND P. TRIGUERO-NAVARRO*

*Instituto de Matematica Multidisciplinar, Universitat Politècnica de València,
València, Spain
E-mail: jrtorre@mat.upv.es

Abstract. In this paper, we present a family of iterative schemes for solving nonlinear systems with 3 real parameters. If we do not fix values for the parameters this family has order 2, but if we fix two of them we obtain order 5. Starting from the fifth-order family, we study different ways of introducing memory, thus obtaining 6 order methods. We also analyze the efficiency indices of the family and of the methods with memory obtained from it, and we compare them with each other, as well as compare them with other known classes of iterative methods with order 5 and 6. Several numerical experiments are carried out to see the behaviour of the discussed methods, including dynamical planes to compare the stability of the different iterative schemes.
Key words: Nonlinear system, fixed point methods, iterative scheme, methods with memory, dynamical planes.
2010 Mathematics Subject Classification: 65H10, 65B99.

1. Introduction

Over time, iterative methods have become more important as they are useful tools for obtaining approximations to solutions α of nonlinear systems $F(x) = 0$, where $F : \mathbb{D} \subset \mathbb{R}^n \to \mathbb{R}^n$, which partake of scientific, engineering and various other models (details can be found in [10, 11, 16, 21]).

What these iterative methods do is to obtain a sequence of approximations, $\{x^{(k)}\}_{k\geq 0}$, from an initial approximation $x^{(0)}$, which, under certain conditions, converge to the solution of $F(x) = 0$, see for example [6, 8]. One of the best known methods is Newton's method, whose expression is:

$$y^{(k)} = x^{(k)} - F'\left(x^{(k)}\right)^{-1} F\left(x^{(k)}\right), \quad k = 1, 2, ... \tag{1.1}$$

being $F'(x^{(k)})$ the Jacobian matrix of F evaluated at $x^{(k)}$.

There are many practical situations where the calculations of Jacobian Matrix are expensive and/or it requires a great deal of time for them to be given or calculated. Therefore, derivative free methods are quite popular for finding the roots of nonlinear equations as well as system of nonlinear equations. One of them is Traub-Steffensen's family [20], which is given by

$$\begin{cases} w^{(k)} = x^{(k)} + \gamma F\left(x^{(k)}\right), \\ y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F\left(x^{(k)}\right), \quad k = 1, 2, ... \end{cases} \tag{1.2}$$

where γ is a non-zero real parameter.

Expression (1.2) can be easily recovered from the well-known Newton's method [20], by replacing the Jacobian matrix to the operator $[w^{(k)}, x^{(k)}; F] \approx F'(x^{(k)})$. For the particular value of $\gamma = 1$ in expression (1.2), then scheme deduce to the well known Steffensen's method [19] for systems of nonlinear equations that was introduced by Samanski in [17], with quadratic order of convergence, for every value of parameter γ.

Many methods have been developed which improve the convergence rate of the Steffensen's method or Steffensen-type methods at the expense of additional evaluations of vector functions, divided difference and changes in the points of iterations. In past and recent years, several higher-order multi-point extension of Steffensen's method or Steffensen-type have been proposed and analyzed in the available literature [1, 12, 15, 18]. All these modifications are in the direction of increasing the local order of convergence with the view of increasing their efficiency indices. Such constructions occasionally possess a better order of convergence and efficiency index.

In this paper, we propose a three-step family that has been obtained using Steffensen's method as starting point. The iterative expression of the parametric family is as follows:

$$\begin{cases} y^{(k)} = x^{(k)} - [w^{(k)}, x^{(k)}; F]^{-1} F(x^{(k)}), \\ z^{(k)} = y^{(k)} - \delta [w^{(k)}, x^{(k)}; F]^{-1} F(y^{(k)}) \\ x^{(k+1)} = z^{(k)} - \beta \left(2I - [w^{(k)}, x^{(k)}; F]^{-1} [z^{(k)}, y^{(k)}; F]\right) [w^{(k)}, x^{(k)}; F]^{-1} F(z^{(k)}), \end{cases} \tag{1.3}$$

where $w^{(k)} = x^{(k)} + \gamma F(x^{(k)})$ and γ is a non-zero real parameter.

The rest of the manuscript is organized as follows. In Section 2, we study the order of convergence of the class of iterative methods without memory (1.3) and we introduce memory to this parametric family when $\beta = 1$ and $\delta = 1$. In Section 3, we analyze the efficiency index and the computational efficiency index of the discussed methods, and we draw some figures to see the behaviour of them. We also compare these indexes with those of other family of order of convergence 5 and 6. In Section 4, we show some numerical experiments to see the performances of the discussed methods and confirm the theoretical results. Some conclusions conclude this paper.

2. Convergence analysis

Let $F : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently differentiable function in a neighbourhood D that contains a solution α of $F(x) = 0$, we consider the divided difference operator

$$[x + h, x; F] = \int_0^1 F'(x + th)dt, \tag{2.1}$$

defined by Genochi-Hermite in [13]. Using the Taylor expansion of $F'(x+th)$ at point x and integrating, we obtain the following development

$$[x+h, x; F] = F'(x) + \frac{1}{2}F''(x)h + \frac{1}{6}F'''(x)h^2 + O(h^3). \tag{2.2}$$

We are going to use this expresion in the proof of the following result.

Theorem 2.1. *Let $F : D \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}^n$ be a sufficiently differentiable function in a neighbourhood D of α such that $F(\alpha) = 0$. We assume that the Jacobian matrix of F evaluated at α is nonsingular. Let γ, β and δ real values. Then, taking an initial estimation $x^{(0)}$ close enough to α, the sequence of iterates $\{x^{(k)}\}$ generated by the proposed family (1.3) converges to α with order 2, and its error equation is:*

$$e_{k+1} = (-1+\beta)(-1+\delta)C_2(I + \gamma F'(\alpha))e_k^2 + O(e_k^3), \tag{2.3}$$

where $C_j = \frac{1}{j}F'(\alpha)^{-1}F^{(j)}(\alpha)$ for $j = 2, 3, \ldots$, $e_k = x^{(k)} - \alpha$ and $F'(\alpha)$ is the Jacobian matrix of F evaluated at α.
Moreover, if $\delta = 1$ and $\beta = 1$, then the order of convergence is 5, and its error equation is

$$e_{k+1} = E\left(I + \gamma F'(\alpha)\right)e_k^5 + O(e_k^6),$$

where

$$E = \left(C_2^2\left(5I + 3\gamma F'(\alpha)\right) + \gamma C_2 F'(\alpha)C_2\left(2 + \gamma F'(\alpha)\right)\right)\left(2C_2^2 + \gamma C_2 F'(\alpha)C_2\right).$$

Proof. First, we consider the Taylor expansion of $F(x^{(k)})$, $F'(x^{(k)})$, $F''(x^{(k)})$ and $F'''(x^{(k)})$ around α:

$$F(x^{(k)}) = F'(\alpha)\left(e_k + C_2e_k^2 + C_3e_k^3 + C_4e_k^4 + C_5e_k^5 + O(e_k^6)\right), \tag{2.4}$$

$$F'(x^{(k)}) = F'(\alpha)\left(I + 2C_2e_k + 3C_3e_k^2 + 4C_4e_k^3 + 5C_5e_k^4 + O(e_k^5)\right), \tag{2.5}$$

$$F''(x^{(k)}) = F'(\alpha)\left(2C_2I + 6C_3e_k + 12C_4e_k^2 + 20C_5e_k^3 + O(e_k^4)\right), \tag{2.6}$$

$$F'''(x^{(k)}) = F'(\alpha)\left(6C_3I + 24C_4e_k + 60C_5e_k^2 + O(e_k^3)\right). \tag{2.7}$$

Applying the Genochi-Hermite formula, we obtain that

$$[w^{(k)}, x^{(k)}; F] = F'(\alpha)\left(I + Y_2e_k + Y_3e_k^2 + Y_4e^3\right) + O\left(e_k^4\right),$$

being

$$\begin{aligned}
Y_2 =& C_2\left(2I + \gamma F'(\alpha)\right), \\
Y_3 =& C_3\left(3I + 3\gamma F'(\alpha)\gamma^2F'(\alpha)^2\right) + \gamma C_2F'(\alpha)C_2, \\
Y_4 =& C_4\left(4I + 6\gamma F'(\alpha) + 4\gamma^2F'(\alpha)^2 + \gamma^2F'(\alpha)C_2F'(\alpha) + \gamma^2F'(\alpha)^2C_2\right) \\
&+ \gamma C_2F'(\alpha)C_3 + 3\gamma C_3F'(\alpha)C_2.
\end{aligned}$$

The inverse of the divided difference operator can be expressed as

$$[w^{(k)}, x^{(k)}; F]^{-1} = (I + X_2e_k + X_3e_k^2 + X_4e_k^3)F'(\alpha)^{-1} + O\left(e_k^4\right),$$

where

$$\begin{aligned} X_2 &= -Y_2, \\ X_3 &= -X_2Y_2 - Y_3, \\ X_4 &= -Y_4 - X_2Y_3 - X_3Y_2. \end{aligned}$$

Then, if we denote $e_{y,k} = y^{(k)} - \alpha$,

$$\begin{aligned} e_{y,k} &= e_k - [w^{(k)}, x^{(k)}; F]^{-1}F(x^{(k)}) \\ &= -(X_2 + C_2)e_k^2 - (X_3 + C_3 + X_2C_2)e_k^3 - (X_4 + C_4 + X_3C_2 + X_2C_3)e_k^4 + O(e_k^5). \end{aligned}$$

Thus, we denote by S_i, $i = 2, 3, 4$,

$$\begin{aligned} S_2 &= -(X_2 + C_2), \\ S_3 &= -(X_3 + C_3 + X_2C_2), \\ S_4 &= -(X_4 + C_4 + X_3C_2 + X_2C_3). \end{aligned}$$

So, $y^{(k)} - \alpha = S_2e_k^2 + S_3e_k^3 + S_4e_k^4 + O(e_k^5)$.
Now, we consider the Taylor expansion of $F(y^{(k)})$ around α:

$$F(y^{(k)}) = F'(\alpha)\left(e_{y,k} + C_2e_{y,k}^2 + O(e_{y,k}^3)\right). \tag{2.8}$$

Thus we denote by Z_i the following expressions,

$$\begin{aligned} Z_3 &= -X_2S_2, \\ Z_4 &= -(X_2S_3 + X_3S_2 + C_2S_2^2). \end{aligned}$$

Then, one has that

$$\begin{aligned} z^{(k)} - \alpha &= e_{y,k} - \delta[w^{(k)}, x^{(k)}; F]^{-1}F(y^{(k)}) \\ &= (1-\delta)S_2e_k^2 + ((1-\delta)S_3 + \delta Z_3)e_k^3 + ((1-\delta)S_4 + \delta Z_4)e_k^4 + O(e_k^5). \end{aligned}$$

We now consider the Taylor expansion of $F'(y^{(k)})$, $F''(y^{(k)})$ and $F'''(y^{(k)})$ around α being $e_{y,k} = y^{(k)} - \alpha$ in the same way as was done for $x^{(k)}$.
Applying the Genochi-Hermite formula we obtain

$$[z^{(k)}, y^{(k)}; F] = F'(\alpha)(I + C_2(2-\delta)S_2e_k^2 + C_2((2-\delta)S_3 + \delta Z_3)e_k^3) + O(e_k^4).$$

Thus we denote by D_i the components of $[z^{(k)}, y^{(k)}; F]$, that is,

$$[z^{(k)}, y^{(k)}; F] = F'(\alpha)(I + D_2e_k^2 + D_3e_k^3) + O(e_k^4),$$

where

$$\begin{aligned} D_2 &= C_2(2-\delta)S_2, \\ D_3 &= C_2((2-\delta)S_3 + \delta Z_3). \end{aligned}$$

Thus,

$$2I - [w^{(k)}, x^{(k)}; F]^{-1}[z^{(k)}, y^{(k)}; F] = I - X_2e_k - (X_3 + D_2)e_k^2 + O(e_k^3).$$

If we denoted by $A = 2I - [w^{(k)}, x^{(k)}; F]^{-1}[z^{(k)}, y^{(k)}; F]$, then it follows that

$$A[w^{(k)}, x^{(k)}; F]^{-1} = (I - (D_2 + X_2^2)e_k^2)F'(\alpha)^{-1} + O(e_k^3).$$

Let us now consider the Taylor expansion of $F(z^{(k)})$ around α

$$F(z^{(k)}) = F'(\alpha)\left(e_{z,k} + C_2 e_{z,k}^2 + O(e_{z,k}^3)\right), \tag{2.9}$$

being $e_{z,k} = z^{(k)} - \alpha$. Then,

$$\begin{aligned} x^{(k+1)} - \alpha =& (1-\beta)e_{z,k} + \beta(D_2 + X_2^2)((1-\delta)S_2 e_k^4 + ((1-\delta)S_3 + \delta Z_3)e_k^5) \\ & - \beta C_2(1-\delta)^2 S_2 S_2 e_k^4 + ((1-\delta)S_2((1-\delta)S_3 + \delta Z_3) \\ & + ((1-\delta)S_3 + \delta Z_3)(1-\delta)S_2)e_k^5 + O(e_k^6). \end{aligned}$$

We distinguish the following different cases for the parameters β and δ:

- If $\delta \neq 1$ and $\beta \neq 1$, then e_{k+1} has order of convergence 2 because $e_{z,k}$ does to.
- If $\delta = 1$ and $\beta \neq 1$, then $e_{z,k}$ has order 3, so e_{k+1} also has order 3.
- If $\delta \neq 1$ and $\beta = 1$,then we get order 4 since the term $e_{z,k}$ cancels out but the terms having e_k^4 do not.
- If $\delta = 1$ and $\beta = 1$, then we are going to obtain the error expression. If we replace these values in the error expression what we get is

$$\begin{aligned} e_{k+1} &= (D_2 + X_2^2)Z_3 e_k^5 + Z_3 e_k^5 + O(e_k^6) \\ &= E\left(I + \gamma F'(\alpha)\right)e_k^5 + O(e_k^6), \end{aligned}$$

where

$$E = \left(C_2^2\left(5I + 3\gamma F'(\alpha)\right) + \gamma C_2 F'(\alpha) C_2\left(2I + \gamma F'(\alpha)\right)\right)\left(2C_2^2 + \gamma C_2 F'(\alpha) C_2\right).$$

Therefore, it is proved that if $\beta = 1$ and $\delta = 1$ the resulting parametric family has order of convergence 5. □

We denoted by S_γ the parametric family (1.3) when $\beta = 1$ and $\delta = 1$.

Now, we want to introduce memory to the family S_γ, in order to increase the order of convergence without adding new functional evaluations. To prove the order of convergence of the methods with memory we use the following Ortega-Rheinboldt's Theorem, which can be found in [13].

Theorem 2.2. *Let ϕ 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, $i = 0, \ldots, m$ 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 ϕ is at least p, where p is the unique positive root of the equation

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

As we can see on the error equation, if $I + \gamma F'(\alpha) = 0$, then the order of convergence will be at least 6. But we don't know α. For this reason we are going to approach $F'(\alpha)$ by an expression that only depends on the previous iterations and their functional evaluations, because we want to increase the order of convergence without increase the number of functional evaluations.

One way to approach $F'(\alpha)$ is by the Kurchatov's operator of divided difference. In this case we are going to choose $\gamma_k = -[2x^{(k)} - y^{(k-1)}, y^{(k-1)}; F]^{-1}$, and by replacing γ in the iterative expression of the family S_γ we get a method with memory denoted by SK_y.

Theorem 2.3. *Let $F : D \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}^n$ be a sufficiently differentiable function in a neighbourhood D of α such that $F(\alpha) = 0$. We assume that $F'(\alpha)$ is nonsingular. Then, taking an initial estimation $x^{(0)}$ close enough to α, the sequence of approximations $\{x^{(k)}\}$ generated by SK_y converges to α with order of convergence 6.*

Proof. From the error equation when $\beta = 1$ and $\delta = 1$ we obtain that

$$e_{k+1} \sim (I + \gamma_k F'(\alpha))e_k^5 + O(e_k^6).$$

Let us consider the Taylor's expansion of $F(y^{(k-1)})$, $F'(y^{(k-1)})$, $F''(y^{(k-1)})$ and $F'''(y^{(k-1)})$ around α in the same way as we do in the previous theorem.

Applying the Genochi-Hermite formula to $A_k = [2x^{(k)} - y^{(k-1)}, y^{(k-1)}; F]$, we obtain

$$A_k = F'(\alpha)\left(I + 2C_2e_k - 2C_3e_{y,k-1}e_k + C_3e_{y,k-1}^2 + 4C_3e_k^2\right) + O_3\left(e_k, e_{y,k-1}\right),$$

where $O_3(e_k, e_{y,k-1})$ denoted all the terms e_k and $e_{y,k-1}$ such that the sum of their exponents is at least 3.

Then, the inverse of the divided difference operator is

$$\begin{aligned} A_k^{-1} =& (I - 2C_2e_k - C_3e_{y,k-1}^2 + 2C_3e_{y,k-1}e_k + 4(C_2^2 - C_3)e_k^2)F'(\alpha)^{-1} \\ &+ O_3\left(e_k, e_{y,k-1}\right). \end{aligned}$$

Thus,

$$\begin{aligned} I + \gamma_k F'(\alpha) =& 2C_2e_k + C_3e_{y,k-1}^2 - 2C_3e_{y,k-1}e_k - 4(C_2^2 - C_3)e_k^2) \\ &+ O_3\left(e_k, e_{y,k-1}\right). \end{aligned}$$

Thus $I + \gamma_k F'(\alpha)$ can have the behaviour of e_k, $e_ke_{y,k-1}$, e_k^2 or $e_{y,k-1}^2$. Obviously, factors $e_ke_{y,k-1}$ and e_k^2 tend to be faster at 0 than e_k, so we have to see whether e_k or $e_{y,k-1}^2$ converges faster.

Let us suppose that the R-order of the method is at least p. Therefore, it is satisfied

$$e_{k+1} \sim D_{k,p}e_k^p,$$

where $D_{k,p}$ tends to the asymptotic error constant, D_p, when $k \to \infty$.

On the other hand, let us suppose that the sequence $\{y^{(k)}\}$ has R-order p_1. Therefore, it is satisfied that

$$e_{y,k} \sim D_{k,p_1}e_k^{p_1},$$

where D_{k,p_1} tends to the asymptotic error constant, D_{p_1}, when $k \to \infty$.

It then follows that

$$\frac{e_k}{e_{y,k-1}^2} \sim \frac{e_{k-1}^p}{e_{k-1}^{2p_1}} \sim e_{k-1}^{p-2p_1}.$$

Thus, if $p > 2p_1$ we have that $\frac{e_k}{e_{y,k-1}^2}$ converges to 0, we have that the behaviour will be that of $e_{y,k-1}^2$, otherwise the behaviour will be as of e_k.

So, if we assume $p > 2p_1$ we have $I + \gamma_k F'(\alpha) \sim e_{y,k-1}^2$. From the error equation and the above relationship it is obtained:

$$e_{k+1} \sim e_{y,k-1}^2 e_k^5 \sim e_{k-1}^{2p_1} e_k^5. \tag{2.10}$$

On the other hand, assuming that the R-order of the method is at least p, one has that

$$e_{k+1} \sim D_{k,p}(D_{k-1,p} e_{k-1}^p)^p = D_{k,p} D_{k-1,p}^p e_{k-1}^{p^2}. \tag{2.11}$$

In the same way that the relation (2.10) is obtained, and supposing that the sequence $y^{(k)}$ has R-order p_1, we obtain that

$$e_{k+1} \sim e_{y,k-1}^2 (e_{k-1}^p)^5 \sim e_{k-1}^{2p1} + e_{k-1}^{5p} \sim e_{k-1}^{5p+2p_1}. \tag{2.12}$$

On the other hand, by the error equation of $e_{y,k}$, we have

$$e_{y,k} \sim (I + \gamma_k F'(\alpha)) e_k^2 \sim e_{k-1}^{2p_1} (e_{k-1}^p)^2 \sim e_{k-1}^{2p+2p_1}. \tag{2.13}$$

Assuming that sequence $\{y^{(k)}\}$ has R-order at least p_1, we have

$$e_{y,k} \sim e_k^{p_1} \sim e_{k-1}^{pp_1}. \tag{2.14}$$

Then, equating the exponents of e_{k-1} on (2.11) and on (2.12), and equating on the other hand the exponents of e_{k-1} on (2.13) and on (2.14) it follows that:

$$\begin{aligned} p^2 &= 5p + 2p_1, \\ pp_1 &= 2p + 2p_1 \end{aligned}$$

whose only positive solution is $p = 6$ and $p_1 = 3$, so it is proved that the order of method SK_y is 6. $\square$

Other ways to introduce memory to the class of iterative schemes when $\beta = 1$ and $\delta = 1$ are:

- If we choose $\gamma_k = -[x^{(k)}, x^{(k-1)}, F]^{-1}$, then substituting the S_γ family parameter for this approximation, yields a method with memory which we denote by SD_x.
- If we choose $\gamma_k = -[2x^{(k)} - x^{(k-1)}, x^{(k-1)}, F]^{-1}$, then substituting the S_γ family parameter for this approximation, yields a method with memory which we denote by SK_x.
- If we choose $\gamma_k = -[x^{(k)}, y^{(k-1)}, F]^{-1}$, then substituting the S_γ family parameter for this approximation, yields a method with memory which we denote by SD_y.
- If we choose $\gamma_k = -[x^{(k)}, z^{(k-1)}, F]^{-1}$, then substituting the S_γ family parameter for this approximation, yields a method with memory which we denote by SD_z.
- If we choose $\gamma_k = -[2x^{(k)} - z^{(k-1)}, z^{(k-1)}, F]^{-1}$, then substituting the S_γ family parameter for this approximation, yields a method with memory which we denote by SK_z.

In the following result, we establish the order of convergence that the previous methods with memory reach.

Theorem 2.4. *Let $F : D \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}^n$ be a sufficiently differentiable function in a neighbourhood D of α such that $F(\alpha) = 0$. We assume that $F'(\alpha)$ is nonsingular. Then, taking an initial estimation $x^{(0)}$ close enough to α, we have*

- *the sequence of iterates $\{x^{(k)}\}$, generated by method SD_x, converges to α with order $p = \frac{5+\sqrt{29}}{2} \approx 5.19$.*
- *the sequence of iterates $\{x^{(k)}\}$, generated by method SK_x, converges to α with order $p = \frac{5+\sqrt{33}}{2} \approx 5.37$.*
- *the sequence of iterates $\{x^{(k)}\}$, generated by method SD_y, converges to α with order $p = 3 + \sqrt{6} \approx 5.449$.*
- *the sequence of iterates $\{x^{(k)}\}$, generated by method SD_z, converges to α with order $p = 3 + \sqrt{7} \approx 5.64$.*
- *the sequence of iterates $\{x^{(k)}\}$, generated by method SK_z, converges to α with order 6.*

3. Efficiency indexes

Next, we study the efficiency index of our parametric family and the memory methods obtained from it. We compare this index with that of the family presented in [2, 3], which we denote by BM.

If d is the number of functional evaluations per iteration and p is the order of convergence of the method, then the efficiency index defined by Ostrowski in [14] is

$$I = p^{1/d}.$$

When we deal with a system $F(x) = 0$ of size $n\ timesn$, n and n^2 functional evaluations per iteration are required to compute F and F', respectively. In addition, $\frac{n^2-n}{2}$ evaluations are required for each divided difference operator. The sum of all functional evaluations is denoted by d.

Our parametric family S_γ performs 3 evaluations of F and calculates 2 divided difference operators, so the number of evaluations is

$$d = n^2 + 2n.$$

We have that in the case of methods with memory obtained from S_γ, we calculate a further divided difference so the number of evaluations would be equal to the above plus $\frac{n^2-n}{2}$, that is, $d = \frac{3}{2}(n^2 + n)$.

The efficiency index only takes into account the number of functional evaluations and the order of the method, but it does not consider the operations (products and quotients) performed to obtain the approximations, for this reason we are also going to study the computational cost of our methods and their computational efficiency index.

The IC or computational efficiency index is calculated in the following way

$$IC = p^{1/(d+op)},$$

being p the order of convergence, d the number of functional evaluations and op the number of computational operations performed. Operations such as product by a scalar or sum of vectors are not taken into account when calculating the operations performed. Multiplying a matrix by a vector requires n^2 operations, which is the same number of operations needed to perform a divided difference operator.

On the other hand, to calculate a linear inverse operator, a system of linear equations of dimension $n \times n$ must be solved, in which an LU decomposition of matrices is calculated and two triangular systems are solved, with a total cost of $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ operations.

However, in order to solve r linear systems with the same matrix of coefficients, the LU decomposition is performed only once, so that the total computational cost is only $\frac{1}{3}n^3 + rn^2 - \frac{1}{3}n$.

Our S_γ family computes 1 matrix-vector product, solves 4 linear systems with the same matrix and computes 2 different divided difference operators, so the number of operations is

$$op = \frac{1}{3}n^3 + 7n^2 - \frac{1}{3}n. \tag{3.1}$$

On the other hand, we have that in the case of the memory methods we calculate one more divided difference and solve a linear system with a different matrix, so the number of operations would be equal to the previous one plus $n^2 + \frac{1}{3}n^3 + n^2 - \frac{1}{3}n$, that is, the number of operations we perform is $op = \frac{2}{3}n^3 + 9n^2 - \frac{2}{3}n$.

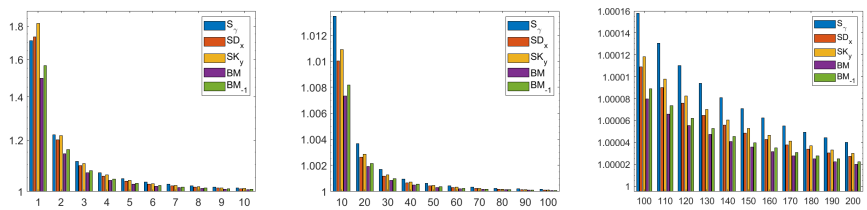
Table ?? shows the number of functional evaluations and the number of operations of the proposed methods and the family BM.

Table 1. Number of functional evaluations and operations, per iteration, of S_γ, SD_*, SK_* and BM

	S_γ	Memory methods	BM
d	$n^2 + 2n$	$\frac{3}{2}(n^2 + n)$	$2(n^2 + n)$
op	$\frac{1}{3}n^3 + 7n^2 - \frac{1}{3}n$	$\frac{2}{3}n^3 + 9n^2 - \frac{2}{3}n$	$n^3 + 6n^2 - n$

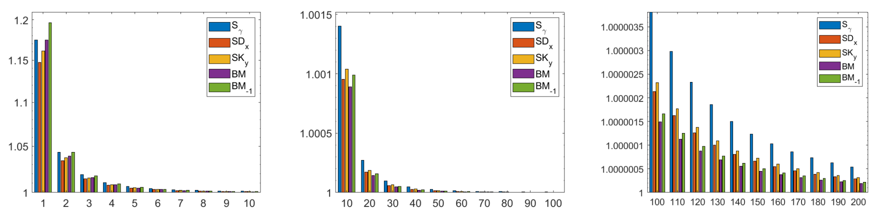
In the following figures, we have plotted the efficiency index and the computational efficiency index for different values of n, where $n \times n$ is the size of the system to be solved. Of the methods with memory we have plotted those that obtain the lowest and the highest index, that is, SD_x is the method that will obtain the lowest index of all those with memory, and the SK_y method, the one with the highest index, although the latter has the same index as SK_z, so we choose only one as a representative.

FIGURE 1. Efficiency index



(A) $n = 1, \dots, 10$ (B) $n = 10, 20, \dots, 100$ (C) $n = 100, 110, \dots, 200$

FIGURE 2. Computational efficiency index



(A) $n = 1, 2, \dots, 10$ (B) $n = 10, 20, \dots, 100$ (C) $n = 100, 110, \dots, 200$

As can be seen in the figures, the methods with the highest efficiency indexes are the methods of family S_γ and method SK_y, which has the same indexes as method SK_z.

4. Numerical experiments

In this section, we compare our class of iterative schemes S_γ and our methods with memory, by solving two classic problems of applied mathematics: the integral equation of Hammerstein and the partial derivative equation of Fisher. We also analyze an academic problem of size 2×2 to represent the dynamical planes of the proposed schemes and we compare them under other point of view. We compare the our schemes with methods belonging to the parametric family introduced in [2] and [3].

For computational calculations, we have use Matlab 2020b with an arithmetic precision of a 200 digits. As stopping criterion we use

$$\|x^{(k+1)} - x^{(k)}\| + \|F(x^{(k+1)})\| < T,$$

where T is the tolerance, which will be different for each method. We also use a maximum of 50 iterations as stopping criterion.

The numerical results we are going to compare the methods in the different examples are

- the approximation obtained $x^{(k+1)}$,
- the value $\|F(x^{(k+1)})\|$,
- the distance between the lasts two iterations $\|x^{(k+1)} - x^{(k)}\|$,
- the number of iterations that have been needed to verify the stopping criterion,
- the computational time,
- and the approximated computational order of convergence (ACOC), defined by Cordero and Torregrosa in [7], which has the following expression

$$p \approx ACOC = \frac{\ln(\|x^{(k+1)} - x^{(k)}\|_2 / \|x^{(k)} - x^{(k-1)}\|_2)}{\ln(\|x^{(k)} - x^{(k-1)}\|_2 / \||x^{(k-1)} - x^{(k-2)}\|_2)}.$$

4.1. **Hammerstein's equation.** In this example, we consider the well-known Hammerstein integral equation (see [13]), which is given as follows:

$$x(s) = 1 + \frac{1}{5}\int_0^1 F(s,t)x(t)^3 dt, \tag{4.1}$$

being $x \in \mathbb{C}[0,1]$, $s,t \in [0,1]$ and the kernel F is

$$F(s,t) = \begin{cases} (1-s)t & t \le s, \\ s(1-t) & s \le t. \end{cases}$$

We transform the above equation into a finite-dimensional problem using the Gauss-Legendre quadrature formula given as $\int_0^1 f(t)dt \approx \sum\limits_{j=1}^{7} \omega_j f(t_j)$, where the nodes t_j and the weights t_j are determined for $n = 7$ by the Gauss-Legendre quadrature formula. In this case, the abscissas and weights are in the following table.

i	Weights ω_i	Nodes t_i
1	0.0647424831	0.0254460438
2	0.1398526957	0.1292344072
3	0.1909150252	0.2970774243
4	0.2089799185	0.5
5	0.1909150252	0.7029225757
6	0.1398526955	0.8707655928
7	0.0647424831	0.9745539561

Denoting the approaches of $x(t_i)$ by x_i ($i = 1, \dots, 7$), the following nonlinear system is obtained

$$5x_i - 5 - \sum_{j=1}^{7} a_{ij} x_j^3 = 0,$$

where $i = 1, \dots, 7$ and

$$a_{ij} = \begin{cases} w_j t_j (1 - t_i) & j \le i, \\ w_j t_i (1 - t_j) & i < j. \end{cases}$$

Starting from an initial approximation

$$x^{(0)} = 0.5(1, \dots, 1)^T,\ x^{(-1)} = y^{(-1)} = z^{(-1)} = 0.4(1, \dots, 1)^T$$

and with a tolerance 10^{-50}, we obtain the approximations for different members of the parametric family, the results of which are contained in Table 2.

TABLE 2. Hammerstein's equation results

Method	$\|x^{(k+1)} - x^{(k)}\|_2$	$\|F(x^{(k+1)})\|_2$	Iteration	ACOC	Time
S_1	1.70931×10^{-190}	2.41129×10^{-951}	4	4.9997	12.9578
S_{-1}	2.1817×10^{-96}	1.46237×10^{-482}	4	4.99841	12.3453
SD_x	1.39146×10^{-243}	3.58645×10^{-1007}	4	5.20041	18.1750
SK_x	2.50789×10^{-257}	2.536×10^{-1007}	4	5.39606	18.3734
SD_y	1.62644×10^{-272}	3.92876×10^{-1007}	4	5.44404	19.5078
SK_y	9.09617×10^{-54}	3.40949×10^{-324}	3	5.75784	14.8047
SD_z	6.53589×10^{-51}	4.11969×10^{-290}	3	5.44555	14.4000
SK_z	2.52685×10^{-54}	8.30666×10^{-329}	3	5.82853	14.8203
BM_{-1}	1.00801×10^{-56}	2.72453×10^{-169}	5	2.99948	15.4484
BM_1	5.63171×10^{-183}	9.85111×10^{-917}	4	4.99976	13.1969

4.2. **Fisher's equation.** In this second example we are going to study the Fisher equation proposed in [9] by Fisher to model the diffusion process in population dynamics. The analytical expression of this equation in partial derivatives is as follows

$$u_t(x,t) = Du_{xx}(x,t) + ru(x,t)\left(1 - \frac{u(x,t)}{p}\right), \qquad x \in [a,b],\ \ t \geq 0, \tag{4.2}$$

where $D \leq 0$ is the diffusion constant, r is the growth level of the species and p is the carrying capacity.

In this case, we will study the Fisher equation for the values $p = 1$, $r = 1$ and $D = 1$ in the interval $[0,1]$ and with the initial condition $u(x,0) = sech^2(\pi x)$ and zero boundary conditions.

We transform the Cauchy problem just described into a set of nonlinear systems by applying an implicit finite difference method, providing the estimated solution at time t_k from the one estimated at t_{k-1}. We denote the spatial step by $h = \frac{1}{n_x}$ and the time step by $k = \frac{T_{max}}{n_t}$, where T_{max} is the final instant and n_x and n_t are the number of subintervals in x and t, respectively. Therefore, we define a mesh of the domain $[0,1] \times [0, T_{max}]$, composed of points (x_i, t_j), as follows

$$x_i = 0 + ih,\ \ i = 0, \dots, n_x, \qquad t_j = 0 + jk,\ \ j = 0, \dots, n_t.$$

Our aim is to approximate the solution of the problem (4.2) at these points of the grid, solving as many nonlinear systems as t_j time nodes in the grid. To do so, we use the following finite differences:

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

We note that for the time step we use first-order backward divided differences and for the spatial step we use second-order centred divided differences. We denote $u_{i,j}$ as the approximation of the solution in (x_i, t_j), and, substituting it into the Cauchy problem, we obtain the system

$$ku_{i+1,j} + (kh^2 - 2k - h^2)u_{i,j} - kh^2 u_{i,j}^2 + ku_{i-1,j} = -h^2 u_{i,j-1},$$

for $i = 1, 2, \ldots, n_x - 1$ and $j = 1, 2, \ldots, n_t$. The unknowns of this system are $u_{1,j}, u_{2,j}, \ldots, u_{n_x-1,j}$, that is, the approximations of the solution at each spatial node for the fixed time t_j.

In this example, we will work with the parameters $T_{max} = 10$, $n_x = 10$ and $n_t = 50$. As we have said, it is necessary to solve as many systems as t_j time nodes, for each of these systems we use the parametric family to approximate its solution. Thus, starting from an initial approximation $u_{i,0} = sech^2(\pi x_i)$, $i = 0, \ldots, n_x$, with a tolerance of 10^{-6}, and $u_{i,-1} = u_{i,0} + 0.5 * ones(1, n_x)$, we obtain the results shows in Table 3.

Table 3. Fisher's equation results

Method	$\|x^{(k+1)} - x^{(k)}\|_2$	$\|F(x^{(k+1)})\|_2$	Iteration	ACOC	Time
S_1	7.39203×10^{-21}	1.2229×10^{-106}	3	5.1619	537.7188
S_{-1}	2.17658×10^{-7}	5.75334×10^{-37}	3	5.6847	359.7344
$S_{-0.1}$	5.8197×10^{-12}	1.63062×10^{-60}	3	5.2641	423.4063
SD_x	4.4694×10^{-28}	1.17155×10^{-147}	3	4.98119	599.0469
SK_x	9.81953×10^{-32}	1.19756×10^{-192}	3	5.95075	545.8906
SD_y	8.19402×10^{-30}	1.11985×10^{-164}	3	5.35656	540.1094
SK_y	9.81953×10^{-32}	1.19756×10^{-192}	3	5.95075	554.4531
SD_z	1.13998×10^{-30}	2.70935×10^{-175}	3	5.54171	558.5781
SK_z	2.4707×10^{-28}	2.19756×10^{-145}	3	5.95075	442.2344
BM_1	1.98612×10^{-17}	1.27687×10^{-88}	3	5.1119	426.4688
BM_{-1}	3.28786×10^{-7}	7.95903×10^{-23}	3	3.1967	345.5469
$BM_{-0.1}$	2.97139×10^{-8}	3.26295×10^{-26}	3	3.1592	406.0781

4.3. **An academic problem.** Next, we carry out the dynamical planes of the iterative methods that we have seen throughout the work and of the BM family with which we compare them. To do so, we start with some preliminary concepts about real dynamics.

The standard form of an iterative method with memory that uses only two previous iterations to calculate the next is:

$$x^{(k+1)} = \phi(x^{(k-1)}, x^{(k)}),\ k \geq 1,$$

being $x^{(0)}$ and $x^{(1)}$ initial estimations. A function defined from $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}^n$ cannot have fixed points. Therefore, an auxiliary vectorial function O is defined by means of

$$O(x^{(k-1)}, x^{(k)}) = (x^{(k)}, x^{(k+1)}) = (x^{(k)}, \phi(x^{(k-1)}, x^{(k)})),\ k = 1, 2, \ldots$$

If $(x^{(k-1)}, x^{(k)})$ is a fixed point of O, then

$$O\left(x^{(k-1)}, x^{(k)}\right) = \left(x^{(k-1)}, x^{(k)}\right),$$

and from the definition of O, we have that

$$\left(x^{(k-1)}, x^{(k)}\right) = \left(x^{(k)}, x^{(k+1)}\right).$$

The basin of attraction of a fixed point x^*, is defined as the set of pre-images of any order such that

$$\mathcal{A}(x^*) = \{y \in \mathbb{R}^n : O^m(y) \to x^*,\ m \to \infty\}.$$

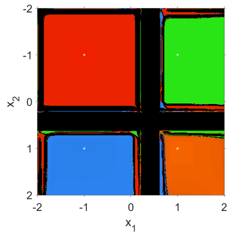
In this case we are going to see the dynamical planes of our methods associated to the following system:

$$\begin{cases} x_1^2 - 1 &= 0, \\ x_2^2 - 1 &= 0. \end{cases}$$

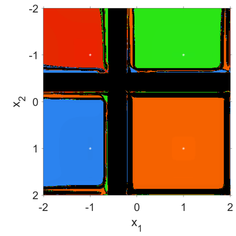
We know that the roots of the polynomial system are $(-1,-1)^T$, $(-1,1)^T$, $(1,-1)^T$ and $(1,1)^T$.

To generate the dynamical planes, we choose a initial point and what we will do is apply our methods taking these point as the initial estimation, see [4, 5]. To choose the point we have generate a mesh of 400×400 points. We have also defined that the maximum number of iterations that each initial estimate must do is 80, and that we will determine that the initial point converges to one of the solutions if the distance to that solution is less than 10^{-3}. We paint in green the initial points that converge to the root $(1,1)^T$, in orange the initial points that converge to the root $(1,-1)^T$, in red the initial points that converge to the root $(-1,1)^T$, in blue the initial points that converge to the root $(-1,-1)^T$ and in black the initial points that do not converge to any root.

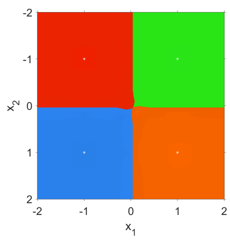
FIGURE 3. Dynamical planes of S_γ and their methods with memory and BM_γ



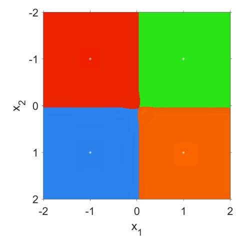
(A) Dynamical plane of $S_{\gamma=1}$



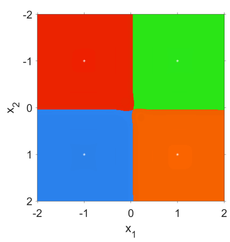
(B) Dynamical plane of $S_{\gamma=-1}$



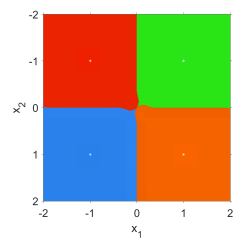
(C) Dynamical plane of SD_x



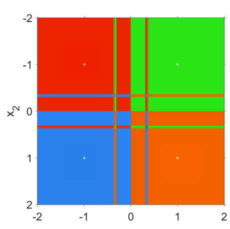
(D) Dynamical plane of SD_y



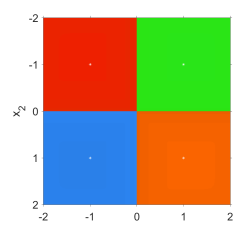
(E) Dynamical plane of SD_z



(F) Dynamical plane of SK_x, SK_y and SK_z



(G) Dynamical plane of $BM_{\gamma=1}$



(H) Dynamical plane of $BM_{\gamma=-1}$

In Table 4, we show the results obtained for this system of equations with initial estimates $x^{(0)} = (1.5, 1.5)^T$ and $x^{(-1)} = y^{(-1)} = z^{(-1)} = (2, 2)^T$ and a tolerance 10^{-100}.

Table 4

Method	$\|x^{(k+1)} - x^{(k)}\|_2$	$\|F(x^{(k)})\|$	Iteration	ACOC	Time
S_1	1.1063×10^{-161}	1.181×10^{-804}	5	4.9999	5.7469
S_{-1}	2.6131×10^{-129}	3.5177×10^{-774}	4	6	2.7297
SD_x	3.98468×10^{-508}	4.91427×10^{-2637}	5	5.1917	3.3453
SK_x	1.25208×10^{-144}	4.25698×10^{-866}	4	6.0	2.7297
SD_y	7.09892×10^{-111}	9.22571×10^{-603}	4	5.44552	6.1703
SK_y	1.25208×10^{-144}	4.25698×10^{-866}	4	6.0	2.5281
SD_z	2.05457×10^{-118}	5.67265×10^{-667}	4	5.6415	4.9922
SK_z	1.25208×10^{-144}	4.25698×10^{-866}	4	6.0	2.5016
BM_1	2.85672×10^{-403}	1.18911×10^{-2014}	5	5.0	6.0812
BM_{-1}	7.69125×10^{-167}	6.82466×10^{-499}	7	3.0	8.4000

5. Conclusions

As we have seen in the work, the parametric family (1.3) has order 2 for any value of β, δ and γ, but if we fix $\beta = 1$ and $\delta = 1$ we get that this resulting parametric family, which we denote by S_γ, has order 5. Using divided difference operators we introduce memory to this parametric family S_γ in different ways. By introducing the memory we have managed to increase the order to 6. We study the efficiency index and the computational efficiency index of our family and our memory methods and compare with other family of order 5 and 6, and we just saw that our family S_γ is better for every size of the system in both indexes. The numerical experiments have confirm the theoretical results and we can observe that our S_γ parametric family takes the fewer computational time in general, but that our methods with memory, especially the SK_y and SK_z methods, obtain better results. Not forgetting that in the academic example, the dynamic planes obtained for the methods with memory present larger basin of convergent initial approximations than those of the S_{gamma} and BM classes.

Acknowledgement. The authors would like to thank the anonymous reviewers for their help to improve the final version of this manuscript.

Financial disclosure. This research was partially supported by Spanish Ministerio de Ciencia, Innovacia y Universidades PGC2018-095896-B-C22, by the internal research project ADMIREN of Universidad Internacional de La Rioja (UNIR) and was partially supported by Universitat Politecnica de Valencia Contrato Predoctoral PAID-01-20-17 (UPV).

References

[1] I. Argyros, S. Regmi, *Undergraduate Research at Cameron University on Iterative Procedures in Banach and Other Spaces*, Nova Science Publisher, 2019.

[2] I. Argyros, D. Sharma, C. Argyros, S. Parhi, S. Sunanda, *A family of fifth and sixth convergence order methods for nonlinear models*, Symmetry, **13**(2021).

[3] R. Behl, E. Martinez, *A new high-order and efficient family of iterative techniques for nonlinear models*, Complexity, (2020).

[4] F. Chicharro, A. Cordero, N. Garrido, N., J.R. Torregrosa, *Stability and applicability of iterative methods with memory*, Journal of Mathematical Chemistry, **57**(2019), 1282-1300.

[5] F. Chicharro, A. Cordero, J.R. Torregrosa, *Drawing dynamical and parameters planes of iterative families and methods*, The Scientific World Journal, **2013**(2013).

[6] A. Cordero, C. Jordan, E. Sanabria, J.R. Torregrosa, *Design, convergence and stability of a fourth-order class of iterative methods for solving nonlinear vectorial problems*, Fractal and Fractional, **5**(2021).

[7] A. Cordero, J.R. Torregrosa, *Variants of Newton's method using fifth-order quadrature formulas*, Appl. Math. Comput., **190**(2007), 686-698.

[8] A. Cordero, E. Villalba, J.R. Torregrosa, P. Triguero-Navarro, *Convergence and stability of a parametric class of iterative schemes for solving nonlinear systems*, Mathematics, **9**(2021).

[9] R. Fisher, *The wave of advance of advantageous genes*, Annals of Eugenics, **7**(1937), 355-369.

[10] C. Grosan, A. Abraham, *A new approach for solving nonlinear equations systems*, IEEE Trans. Syst. Man Cybernet Part A: Syst. Humans, **38**(2008), 698-714.

[11] J.J. Moré, *A collection of nonlinear model problems*, in: E.L. Allgower, K. Georg (Eds.), Computational Solution of Nonlinear Systems of Equations, Lectures in Applied Mathematics, vol. **26**, Amer. Math. Soc. Providence, RI, 1990, 723-762.

[12] M. Narang, S. Bathia, A.S. Alshomrani, V. Kanwar, *General efficient class of Steffensen type methods with memory for solving systems on nonlinear equations*, J. Comput. Appl. Math., **352**(2019), 23-39.

[13] J. Ortega, W. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New-York, 1970.

[14] A.M. Ostrowski, *Solutions of Equations and System of Equations*, Academic Press, New York, 1960.

[15] M.S. Petkovíc, J.R. Sharma, *On some efficient derivative-free iterative methods with memory for solving systems of nonlinear equations*, Numer. Algor., **71**(2016), 457-474.

[16] S. Regmi, *Optimized Iterative Methods with Applications in Diverse Disciplines*, Nova Science Publisher, 2021.

[17] V. Samanskii, *On a modification of the Newton method*, Ukrain. Math., **19**(1967), 133-138.

[18] J.R. Sharma, H. Arora, *Efficient higher order derivative-free multipoint methods with and without memory for systems of nonlinear equations*, Int. J. Comput. Math., **95**(2018), 920-938.

[19] J. Steffensen, *Remarks on iteration*, Skand. Aktuarietidskr, **16**(1933), 64-72.

[20] J. Traub, *Iterative Methods for the Solution of Equations*, Prentice-Hall, 1964.

[21] I.G. Tsoulos, A. Stavrakoudis, *On locating all roots of systems of nonlinear equations inside bounded domain using global optimization methods*, Nonlinear Anal. Real World Appl., **11**(2010), 2465-2471.

Received: December 2, 2021; Accepted: October 5, 2022.