




# Simultaneous roots for vectorial problems

Francisco Chinesta<sup>1</sup> · Alicia Cordero<sup>2</sup> · Neus Garrido<sup>2</sup> · Juan R. Torregrosa<sup>2</sup>  · Paula Triguero-Navarro<sup>2</sup>

Received: 20 February 2023 / Revised: 6 June 2023 / Accepted: 8 June 2023 /  
Published online: 23 June 2023  
© The Author(s) 2023

## Abstract

In this manuscript, we design an iterative step that can be added to any numerical process for solving systems of nonlinear equations. By means of this addition, the resulting iterative scheme obtains, simultaneously, all the solutions to the vectorial problem. Moreover, the order of this new iterative procedure duplicates that of their original partner. We apply this step to some known methods and analyse the behaviour of these new algorithms, obtaining simultaneously the roots of several nonlinear systems.

**Keywords** Iterative procedures · Nonlinear equations and systems · Simultaneous · Dynamical analysis

**Mathematics Subject Classification** 65H05 · 65H10

## 1 Introduction

In applied sciences, we need to solve a system of equations, and in many cases these systems are nonlinear. We cannot always solve this type of systems in an exact way, due to the complexity of the problem. For this reason, we obtain an approach to the solution instead of

---

Communicated by Baisheng Yan.

---

✉ Juan R. Torregrosa  
jrtorre@mat.upv.es

Francisco Chinesta  
Francisco.CHINESTA@ensam.eu

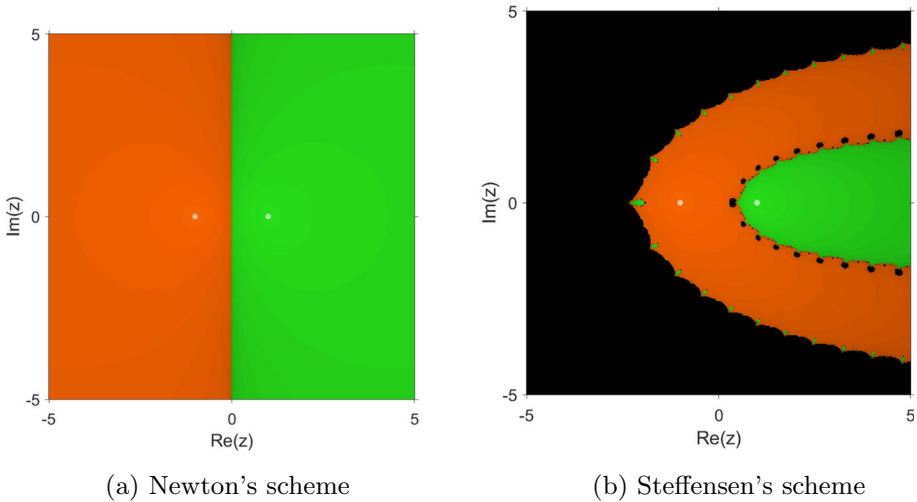
Alicia Cordero  
acordero@mat.upv.es

Neus Garrido  
neugarsa@mat.upv.es

Paula Triguero-Navarro  
ptrinav@upvnet.upv.es

<sup>1</sup> PIMM Lab and ESI Chair, Arts et Metiers Institute of Technology, Paris, France

<sup>2</sup> Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, València, Spain



**Fig. 1** Newton's and Steffensen's dynamical planes for  $p(x)$

solving it exactly. These systems are described as  $F(x) = 0$ , where  $F : D \subseteq \mathbb{R}^m \rightarrow \mathbb{R}^r$  is a vectorial function of  $m$  variables.

One way to obtain these approximations is by using iterative methods. Iterative methods create a sequence of approximations starting from a seed that, under specific conditions, tends to the solution.

But, what if instead of searching for only one of the solutions, we aim to obtain more? One option would be trying to obtain different solutions by using several different initial approximations. But with this we have a problem, what happens if both estimates converge to the same root? In the following, we are going to show this problem by representing the dynamical planes for Newton's and Steffensen's methods to illustrate what we have just mentioned. What we observe in these cases is if the initial points converge or not to the roots of our problems. In this case, both methods are applied to  $p(x) = x^2 - 1$ , with roots 1 and  $-1$ .

We chose a mesh of  $400 \times 400$  points to create the dynamical planes, and what we do then is apply our methods to each one of these points, taking it as an initial estimation. The  $X$  and  $Y$  axes correspond to the real and imaginary part of the initial estimate, respectively. In addition, we set the maximum number of iterations to 80, and we consider the convergence of the initial point to a solution if the distance to that root is less than  $10^{-3}$ . In orange, we mark the initial points that converge to  $-1$ , in green those that converge to 1, and in black those that do not converge to any root.

In this case, if we take in Fig. 1 two different initial approximations and both are in the same basin of attraction, we obtain two sequences converging to the same solution.

Therefore, if we design a method that calculates simultaneously as many sequences of approximations as solutions the problem has, we will avoid this problem. This is due to the process of calculating an iterate takes into account who are the other iterates.

Schemes to find simultaneously roots of scalar equations are usually defined for polynomial functions. They generate, starting with an initial set of estimations, sequences of approximations that simultaneously tend to several roots. See, for instance (Petković et al.

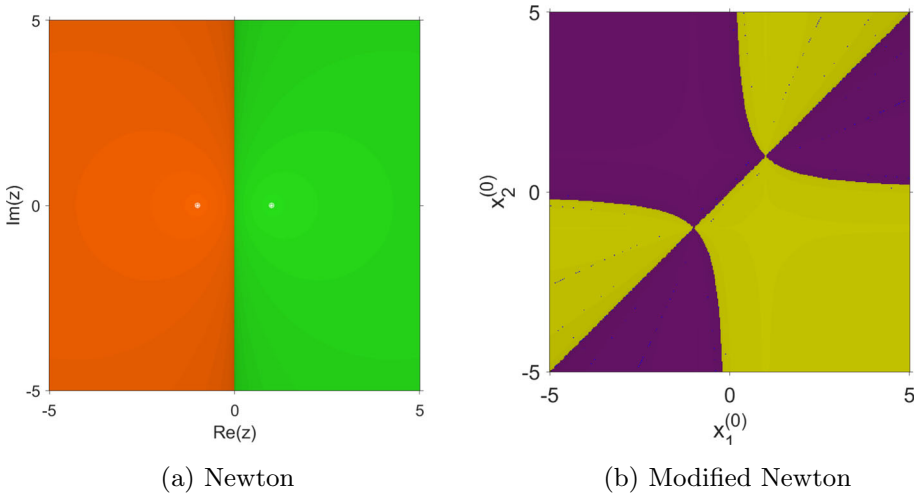


Fig. 2 Dynamical plane of Newton and modified Newton schemes

2014; Proinov 2015; Proinov and Vassileva 2016; Petković et al. 1997; Proinov and Ivanov 2019; Proinov and Vassileva 2019) and their references inside.

From Ehrlich’s method (Ehrlich 1967), the authors in Cordero et al. (2022), design an iterative procedure that can be added to any root-finding scheme for solving arbitrary nonlinear functions. In addition, if the original iterative method has order of convergence  $p$ , the new scheme will duplicate its order to  $2p$ . In addition, the new procedure can be used for obtaining the solutions of the equation simultaneously.

Next, we build the dynamical planes of the modified Newton’ and Steffensen’s methods, applied to the same equation with the same convergence criterion. In this case, one axis corresponds to one component of the initial estimation  $x_1^{(0)}$ , being the other axis component  $x_2^{(0)}$ . We represent the initial point in purple if the component of the point on the  $x_1^{(0)}$  converges to the root  $-1$  and the component in the axis  $x_2^{(0)}$  converges to the root  $1$ . We represent the initial point in yellow if the part of the point on the  $x_1^{(0)}$  converges to the root  $1$  and the part in the axis  $x_2^{(0)}$  converges to the root  $-1$ . If there is no convergence, we mark the starting point in blue.

In Fig. 2, we show Newton’s and modified Newton’s procedures applied in the polynomial dynamical planes. The basins of attraction cover the entire space in case of Newton’s scheme, and also for its variant for finding roots simultaneously.

In Fig. 3, dynamical planes are shown for Steffensen’ and modified Steffensen’s methods. In this case, as we can see, there is no convergence for Steffensen’s method in some black zones, for example at  $z = -5$ , but its variant converges to the roots at any point, except a small zone where  $x_1^{(0)} = x_2^{(0)} = 0$ .

As we can see in the dynamical planes of the modified Newton’ and Steffensen’s methods, taking suitable initial estimations  $x_1^{(0)} = 2$  and  $x_2^{(0)} = 5$  we converge to both roots. This can not be achieved by iterating both non simultaneous methods in parallel, since both were in the basin of the attraction of the root  $1$ .

In this paper, we generalize this idea to nonlinear systems, thus designing an iterative procedure that can be added to any iterative method for solving systems of nonlinear equations.

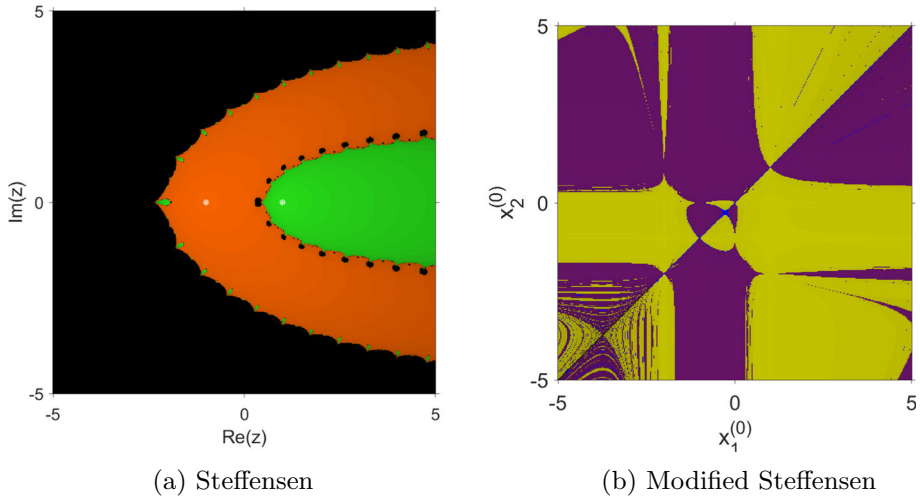


Fig. 3 Dynamical plane of Steffensen and modified Steffensen

Indeed, the new resultant method duplicates the order of convergence and can get several solutions simultaneously.

In Sect. 2, we design the mentioned step and study the order of convergence of the resulting method. In the third Section, we carry out some numerical experiments with known iterative procedures to which we add the simultaneous step, to see the performance of the new schemes. We finish the work in Sect. 4 with some conclusions, derived from the study.

### 2 Convergence analysis

Let  $F(x) = 0, F : \mathbb{R}^m \rightarrow \mathbb{R}^r$ , be a system of nonlinear equations, where the number of unknowns is  $m$  and the number of equations is  $r$ . Note that the system is written as a column vector of size  $r \times 1$ .

Suppose that this system has  $n$  solutions, which are denoted by

$$\alpha_i = (\alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_m}) \text{ for } i = 1, \dots, n.$$

We design an iterative scheme to obtain all the solutions simultaneously. Therefore, we take a set of  $n$  initial approximations which we denote by  $x_i^{(0)} = (x_{i_1}^{(0)}, x_{i_2}^{(0)}, \dots, x_{i_m}^{(0)})$ ,  $i = 1, \dots, n$ . If we define

$$\frac{1}{x_i^{(k)} - x_j^{(k)}} := \left( \frac{1}{x_{i_1}^{(k)} - x_{j_1}^{(k)}}, \frac{1}{x_{i_2}^{(k)} - x_{j_2}^{(k)}}, \dots, \frac{1}{x_{i_m}^{(k)} - x_{j_m}^{(k)}} \right),$$

then we design the iterative step as:

$$\left\{ x_i^{(k+1)} = x_i^{(k)} - \left( F'(x_i^{(k)}) - F(x_i^{(k)}) \sum_{j \neq i} \frac{1}{x_i^{(k)} - x_j^{(k)}} \right)^{-1} F(x_i^{(k)}), \right. \quad (1)$$

for  $i = 1, \dots, n$  and  $k = 0, 1, \dots$

As we can see, the size of matrix  $F'(x_i^{(k)})$  is  $r \times m$  which matches with the size of the product of  $F(x_i^{(k)})$  and  $\frac{1}{x_i^{(k)} - x_j^{(k)}}$ ,  $j \neq i$ , as the above vectors are a column vector of size  $r \times 1$  and a row vector of size  $1 \times m$ , respectively.

We denote this method by *PS*. Now, we prove that this iterative procedure has quadratic order.

**Theorem 1** *Let  $F : \mathbb{R}^m \rightarrow \mathbb{R}^r$  be an enough differentiable function in a neighbourhood of  $\alpha_i$ ,  $D_i \subset \mathbb{R}^m$ . Let us also assume that  $F(\alpha_i) = 0$ , for  $i = 1, \dots, n$  and  $F'(\alpha_i)$  is nonsingular for  $i = 1, \dots, n$ . Then, by taking an estimation  $x_i^{(0)}$  close enough to  $\alpha_i$ ,  $i = 1, \dots, n$ , sequence  $\{x_i^{(k)}\}$  of approximations generated by the *PS* scheme converges to  $\alpha_i$  with order 2.*

**Proof** We denote by  $F = (F_1, F_2, \dots, F_r)$ , where  $F_p : \mathbb{R}^m \rightarrow \mathbb{R}$ , for  $p = 1, 2, \dots, r$ . Now, we consider the Taylor expansion of  $F_p(x_i^{(k)})$  around  $\alpha_i$  for  $p = 1, 2, \dots, r$ ,

$$F_p(x_i^{(k)}) = \sum_{j_1=1}^m \frac{\partial F_p(\alpha_i)}{\partial x_{j_1}} e_{i,kj_1} + \sum_{j_1=1}^m \sum_{j_2=1}^m \frac{\partial^2 F_p(\alpha_i)}{\partial x_{j_1} \partial x_{j_2}} e_{i,kj_1} e_{i,kj_2} + O_3(e_{i,k}), \tag{2}$$

where  $e_{i,kj_1} = x_{ij_1}^{(k)} - \alpha_{ij_1}$  for  $j_1 \in \{1, 2, \dots, m\}$  and  $i \in \{1, \dots, n\}$ , and where  $O_3(e_{i,k})$  the elements where the sums of the exponents of  $e_{i,kj_1}$  is greater than or equal to 3 with  $j_1 \in \{1, 2, \dots, m\}$ .

If we derive this with respect to the variable  $x_{iq}$ , with  $q = 1, 2, \dots, m$ , we get

$$\frac{\partial F_p(x_i^{(k)})}{\partial x_{iq}} = \frac{\partial F_p(\alpha_i)}{\partial x_{iq}} + \sum_{j_1=1}^m \frac{\partial^2 F_p(\alpha_i)}{\partial x_{iq} \partial x_{j_1}} e_{i,kj_1} + O_2(e_{i,k}). \tag{3}$$

Then,

$$\begin{aligned} & \frac{\partial F_p(x_i^{(k)})}{\partial x_{iq}} - F_p(x_i^{(k)}) \sum_{j \neq i}^m \frac{1}{x_{iq} - x_{jq}} = \frac{\partial F_p(\alpha_i)}{\partial x_{iq}} + \sum_{j_1=1}^m \frac{\partial^2 F_p(\alpha_i)}{\partial x_{iq} \partial x_{j_1}} e_{i,kj_1} \\ & - \sum_{j_1=1}^m \frac{\partial F_p(\alpha_i)}{\partial x_{j_1}} e_{i,kj_1} \sum_{j \neq i}^m \frac{1}{e_{i,kq} - e_{j,kq} + \alpha_{iq} - \alpha_{jq}} \\ & + O_2(e_{i,k}). \end{aligned}$$

Then, we can rewrite the last equation as

$$\frac{\partial F_p(x_i^{(k)})}{\partial x_{iq}} - F_p(x_i^{(k)}) \sum_{j \neq i}^m \frac{1}{x_{iq} - x_{jq}} = \frac{\partial F_p(\alpha_i)}{\partial x_{iq}} + \sum_{j_1=1}^m A_{i_q,j_1} e_{i,kj_1} + O_2(e_{i,k}),$$

being

$$A_{i_q,j_1} := \left( \frac{\partial^2 F_p(\alpha_i)}{\partial x_{iq} \partial x_{j_1}} - \frac{\partial F_p(\alpha_i)}{\partial x_{j_1}} \sum_{j \neq i}^m \frac{1}{e_{i,kq} - e_{j,kq} + \alpha_{iq} - \alpha_{jq}} \right),$$

for  $j_1 \in \{1, 2, \dots, m\}$ .

If we write  $A_{i_q} := (A_{i_q,1}, A_{i_q,2}, \dots, A_{i_q,m})$ , and then write  $A$  which rows are  $A_{i_1}, A_{i_2}, \dots, A_{i_m}$  we obtain that

$$F'(x_i^{(k)}) - F(x_i^k) \sum_{j \neq i} \frac{1}{x_i^k - x_j^k} = F'(\alpha_i) (I + F'(\alpha_i)^{-1} A e_{i,k}) + O_2(e_{i,k}).$$

From the above relationship, it follows that

$$\left( F'(x_i^{(k)}) - F(x_i^k) \sum_{j \neq i} \frac{1}{x_i^k - x_j^k} \right)^{-1} = (I - F'(\alpha_i)^{-1} A e_{i,k}) F'(\alpha_i)^{-1} + O_2(e_{i,k}).$$

If we apply the above relation, then  $e_{i,k+1}$  has the following form

$$\begin{aligned} e_{i,k+1} &= e_{i,k} - (I - F'(\alpha_i)^{-1} A e_{i,k}) (e_{i,k} + C_2 e_{i,k}^2) + O_3(e_{i,k}) \\ &= (F'(\alpha_i)^{-1} A - C_2) e_{i,k}^2 + O_3(e_{i,k}). \end{aligned}$$

It is, therefore, proven that  $PS$  has convergence order 2. □

Let  $\phi(x)$  be the fixed point of an iterative method. We define  $PS_\phi$  as follows

$$\begin{cases} x_i^{(k+1)} = \phi(x_i^{(k)}), \\ x_i^{(k+2)} = PS(x_1^{(k+1)}, \dots, x_n^{(k+1)}), \end{cases} \text{ for } i = 1, \dots, n \text{ and } k = 0, 1, \dots$$

that is, an iterative scheme in which we first perform method described by  $\phi$  and then the  $PS$  scheme.

**Theorem 2** *Let  $F : \mathbb{R}^m \rightarrow \mathbb{R}^l$  be a sufficiently differentiable function in a neighbourhood  $D_i \subset \mathbb{R}^m$ , of  $\alpha_i$ . Let us assume that  $F(\alpha_i) = 0$ , for  $i = 1, \dots, n$  and  $F'(\alpha_i)$  is nonsingular for  $i = 1, \dots, n$ . Then, by taking  $x_i^{(0)}$ , an seed close enough to  $\alpha_i$ , for  $i = 1, \dots, n$ , the sequence of iterates  $\{x_i^{(k)}\}$  generated by  $PS_\phi$  method converges to  $\alpha_i$  with order  $2p$ , where  $p$  is the order of convergence of scheme described by  $\phi$ .*

**Proof** By Theorem 1,

$$e_{i,k+2} = (F'(\alpha_i)^{-1} A - C_2) e_{i,k+1}^2 + O_3(e_{i,k+1}). \tag{4}$$

Since we have that  $\phi$  has order  $p$ , this means that  $e_{i,k+1} \sim e_{i,k}^p$ . Substituting the last relation into the equation (4) we obtain that

$$e_{i,k+2} \sim e_{i,k+1}^2 \sim (e_{i,k}^p)^2 \sim e_{i,k}^{2p}.$$

Thus, it is proven that procedure  $PS_\phi$  has order  $2p$ . □

### 3 Numerical experimentation

We use Matlab R2021b with arithmetic precision of one thousand digits for the computational calculations. As a stopping criterion we compare the mean of the norm of the function evaluated at the last iterations with a tolerance of  $10^{-50}$ , that is, if we try to find  $n$  solutions,

$$\frac{1}{n} \sum_{i=1}^n \|F(x_i^{(k+1)})\| < 10^{-50}.$$

We denote by

$$\|F(x^{(k+1)})\| := \frac{1}{n} \sum_{i=1}^n \|F(x_i^{(k+1)})\|$$

and  $x^{(k+1)} = (x_1^{(k+1)}, x_2^{(k+1)}, \dots, x_n^{(k+1)})$ .

We also use a maximum of 100 iterations as a stopping criterion.

For the numerical experiments, we use three different algorithms. The first one is the method *PS*, that we have defined in (1). The others ones are *PS<sub>N</sub>* and *PS<sub>N2</sub>* that are the composition of *PS* with *N* and *N2*, where *N* denote Newton’s scheme and *N2* denote Newton’s method composed with itself, that is

$$\begin{cases} x_i^{(k+1)} = x_i^{(k)} - F'(x_i^{(k)})^{-1} F(x_i^{(k)}), \\ x_i^{(k+2)} = x_i^{(k+1)} - F'(x_i^{(k+1)})^{-1} F(x_i^{(k+1)}), \text{ for } i = 1, \dots, n \text{ and } k = 0, 1, \dots \end{cases}$$

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

- the obtained approximation,
- the mean of the norm of the system evaluated in that set of approximations,
- the distance between the last two sets of iterates,
- the number of iterations needed to achieve the tolerance,
- the computational time and the ACOC (approximate computational convergence order), defined by Cordero and Torregrosa in Cordero and Torregrosa (2007), defined as follows

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

We are going to solve the following nonlinear problems.

**Example 1** We aim to find the intersection points of the circle centred at (0, 0) with radius  $\sqrt{2}$  and the ellipse  $\{(x, y) \in \mathbb{R}^2 : 3x^2 + 2xy + 3y^2 = 5\}$ , that is, find the points (x, y) that are solutions of the following system denoted by *S*(x, y):

$$\begin{aligned} x^2 + y^2 &= 2, \\ 3x^2 + 2xy + 3y^2 &= 5. \end{aligned}$$

The exact solutions of this problem are

$$\left\{ \left( \sqrt{1 + \frac{\sqrt{3}}{2}}, \frac{-1}{\sqrt{4+2\sqrt{3}}} \right), \left( -\sqrt{1 + \frac{\sqrt{3}}{2}}, \frac{1}{\sqrt{4+2\sqrt{3}}} \right), \left( \sqrt{1 - \frac{\sqrt{3}}{2}}, \frac{-1}{\sqrt{4-2\sqrt{3}}} \right), \left( -\sqrt{1 - \frac{\sqrt{3}}{2}}, \frac{1}{\sqrt{4-2\sqrt{3}}} \right) \right\}.$$

Then, we take as initial estimations  $x_1^{(0)} = (1, -0.5)$ ,  $x_2^{(0)} = (-1, 0.5)$ ,  $x_3^{(0)} = (0.5, -1)$  and  $x_4^{(0)} = (-0.5, 1)$ .

As we can see in Table 1, all the methods obtain good results for the chosen initial points. The approximate computational convergence order coincides with the theoretical one or is greater than that one.

**Example 2** The second academical problem to be solved is to obtain all the critical points of function  $g(x, y) = \frac{1}{3}x^3 + y^2 + 2xy - 6x - 3y + 4$ .

**Table 1** Output for intersection of ellipse and circle

Scheme	$\ x^{(k+1)} - x^{(k)}\ $	$\ S(x^{(k+1)})\ $	Iteration	ACOC
$PS$	$2.1809 \times 10^{-39}$	$1.5874 \times 10^{-77}$	8	1.9993
$PS_N$	$1.6021 \times 10^{-94}$	$5.635 \times 10^{-188}$	4	3.5975
$PS_{N2}$	$2.5749 \times 10^{-45}$	$2.6135 \times 10^{-362}$	3	9.2015

**Table 2** Output of the equation  $\nabla g(x, y) = 0$

Scheme	$\ x^{(k+1)} - x^{(k)}\ $	$\ \nabla g(x^{(k+1)})\ $	Iteration	ACOC
$PS$	$3.7423 \times 10^{-40}$	$5.5804 \times 10^{-80}$	8	2.0003
$PS_N$	$6.7853 \times 10^{-61}$	$1.151 \times 10^{-121}$	4	4.0
$PS_{N2}$	$2.3299 \times 10^{-30}$	$8.2801 \times 10^{-244}$	3	8.2457

**Table 3** Output of the system  $F(x) = 0$

Method	$\ x^{(k+1)} - x^{(k)}\ $	$\ F(x^{(k+1)})\ $	Iteration	ACOC
$PS$	$3.637 \times 10^{-4}$	$3.3385 \times 10^{-7}$	17	2.0777
$PS_N$	$4.2398 \times 10^{-4}$	$4.5234 \times 10^{-7}$	3	3.2312
$PS_{N2}$	$6.605 \times 10^{-5}$	0	2	-

To calculate the critical points, we solve  $\nabla g(x, y) = 0$ .

As initial approximations we take  $x_1^{(0)} = (0, 1)$  and  $x_2^{(0)} = (2, -1)$ .

In this case, for all methods, vectors  $(-1, \frac{5}{2})$  and  $(3, -\frac{3}{2})$  are obtained as approximations to the solutions.

As we can see in Table 2, all the methods obtain good results for the chosen initial points. The approximate computational convergence order matches the theoretical one.

**Example 3** Now, we solve the following nonlinear system with size  $200 \times 200$

$$F_i(x) = (x_i^2 - 1)(x_{i+1}^2 - 1) = 0, \quad i = 1, \dots, 199$$

$$F_{200}(x) = (x_1^2 - 1)(x_{200}^2 - 1) = 0,$$

where  $x = (x_1, x_2, \dots, x_{199}, x_{200}) \in \mathbb{R}^{200}$ .

This system has infinite solutions, but we aim to find two of them. As initial points we choose  $x_1^{(0)} = 0.8(1, 1, \dots, 1)$  and  $x_2^{(0)} = -0.8(1, 1, \dots, 1)$ .

In this case, we use Matlab2021b with arithmetic precision of 10 digits and  $10^{-5}$  as tolerance. As we can see in Table 3, the method that does not use a predictor needs 17 iterations to achieve that tolerance.

We conclude that in many cases is very useful to use a predictor scheme, due to the fact that  $PS_{N2}$  obtains the exact solution in only 2 iterations. It is clear that methods with a higher order of convergence converge faster.



## 4 Outcomes and conclusions

In this paper, we define an iterative step that can be added to any iterative scheme for nonlinear systems obtaining a new iterative method that simultaneously find several roots, which has twice the order of the original scheme.

Different known iterative procedures were selected to add this step, and some experiments were carried out numerically to confirm the good properties of the new iterative procedures.

**Acknowledgements** The authors would like to thank the anonymous reviewer for his/helpful comments and suggestions, that have improved the final version of this manuscript.

**Author contributions** All authors contributed equally to this manuscript.

**Funding** Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This research was partially supported by Universitat Politècnica de València, Ayuda Movilidad Estancia Doctorado 01/10/2021 and Universitat Politècnica de València Contrato Predoctoral PAID-01-20-17 (UPV).

**Availability of data and materials** Not applicable.

**Code availability** Not applicable.

## Declarations

**Conflict of interest** All authors declare no conflict of interest.

**Ethics approval** Not applicable.

**Consent to participate and for publication** All authors give their consent to participate and to publication.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Chicharro FI, Cordero A, Garrido N, Torregrosa JR (2020) On the improvement of the order of convergence of iterative methods for solving nonlinear systems by means of memory. *Appl Math Lett* 104:106277. <https://doi.org/10.1016/j.aml.2020.106277>
- Cordero A, Torregrosa JR (2007) Variants of Newton's method using fifth-order quadrature formulas. *Appl Math Comput* 190:686–698. <https://doi.org/10.1016/j.amc.2007.01.062>
- Cordero A, Garrido N, Torregrosa JR, Triguero-Navarro P (2022) Iterative schemes for finding all roots simultaneously of nonlinear equations. *Appl Math Lett* 134:108325. <https://doi.org/10.1016/j.aml.2022.108325>
- Ehrlich LW (1967) A modified newton method for polynomials. *Commun ACM* 10:107–108
- Petković I, Ilić S, Tričković S (1997) A family of simultaneous methods zero-finding methods. *Comput Math Appl* 34:49–59. [https://doi.org/10.1016/S0898-1221\(97\)00206-X](https://doi.org/10.1016/S0898-1221(97)00206-X)
- Petković M, Petković LD, Dzunic J (2014) On an efficient simultaneous method for finding polynomial zeros. *Appl Math Lett* 28:60–65. <https://doi.org/10.1016/j.aml.2013.09.011>
- Proinov P (2015) On the local convergence of ehrlich method for numerical computation of polynomial zeros. *Calcolo* 53. <https://doi.org/10.1007/s10092-015-0155-y>

- Proinov P, Ivanov S (2019) Convergence analysis of sakurai-torii-sugiura iterative method for simultaneous approximation of polynomial zeros. *Comput Appl Math* 357:56–70. <https://doi.org/10.1016/j.cam.2019.02.021>
- Proinov P, Vassileva M (2016) On a family of weistrass-type root-finding methods with accelerated convergence. *Appl Math Comput* 273:957–968. <https://doi.org/10.1016/j.amc.2015.10.048>
- Proinov P, Vassileva M (2019) On the convergence of high-order gargantini-farmer-loizon type iterative methods for simultaneous approximation of polynomial zeros. *Appl Math Comput* 361:202–214. <https://doi.org/10.1016/j.amc.2019.05.026>
- Robinson RC (2012) *An introduction to dynamical systems: continuous and discrete*. American Mathematical Society, Providence
- Shams M, Rafiq N, Kausar N, Agarwal P, Park C, Mir N (2021) On iterative techniques for estimating all roots of nonlinear equation and its system with application in differential equation. *Adv Differ Equ* 2021:480–498. <https://doi.org/10.1186/s13662-021-03636-x>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.