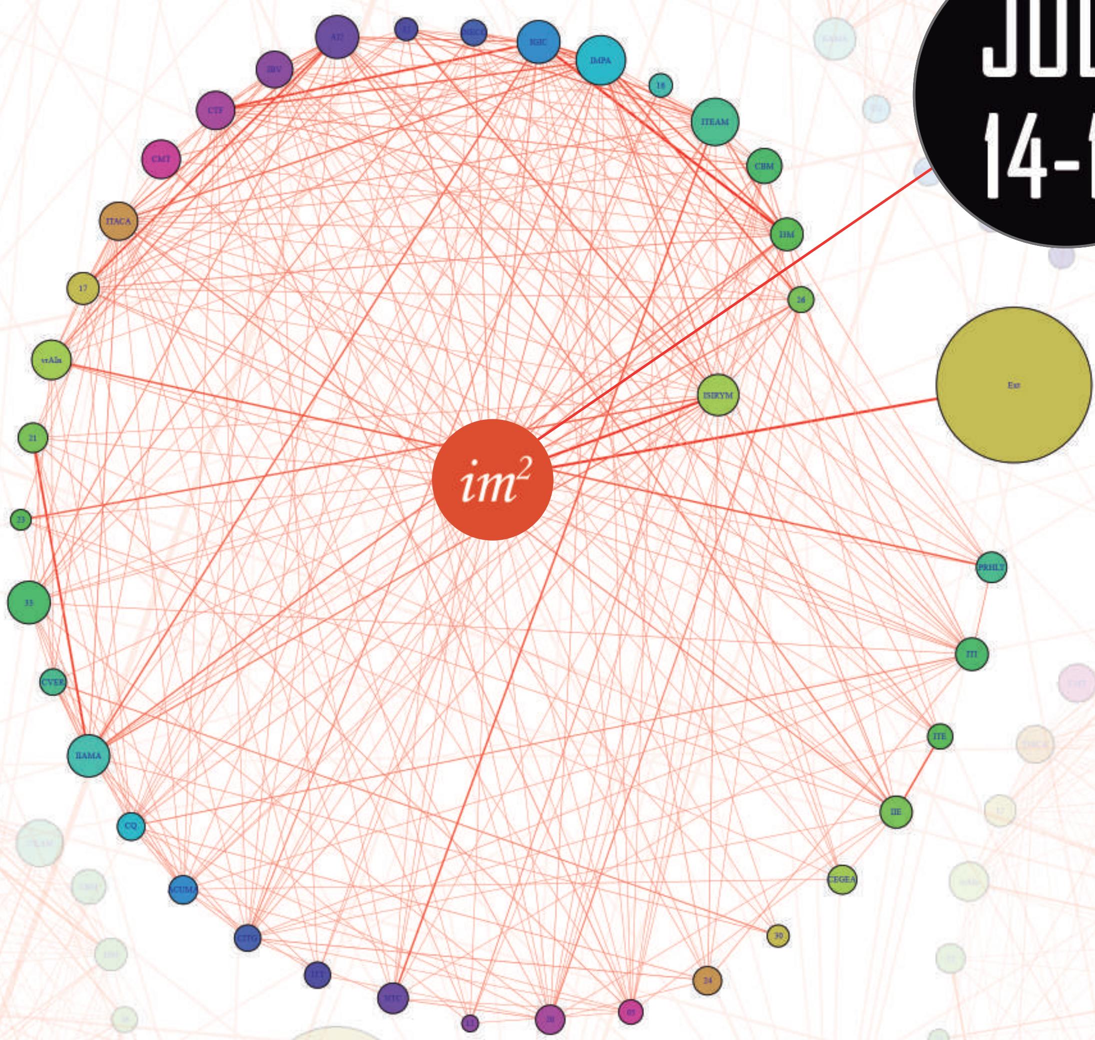
MODELLING FOR ENGINEERING & HUMAN BEHAVIOUR 2021

JULY 14-16



Edited by

Juan Ramón Torregrosa   Juan Carlos Cortés   Antonio Hervás   Antoni Vidal   Elena López-Navarro

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

im²
Instituto Universitario de Matemática Multidisciplinar

I.S.B.N.: 978-84-09-36287-5

November 30th, 2021
Report any problems with this document to imm@imm.upv.es.

Edited by: I.U. de Matemàtica Multidisciplinar, Universitat Politècnica de València.
J.R. Torregrosa, J-C. Cortés, J. A. Hervás, A. Vidal-Ferràndiz and E. López-Navarro

Instituto Universitario
de Matemática Multidisciplinar

Contents

Stability comparison of self-accelerating parameter approximation on one-step iterative methods

F.I. Chicharro ♮, N. Garrido ♭ [1], D. Pérez ♭ and Í. Sarría ♭

(♮) Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera, s/n, 46022-València, Spain,

(♭) Escuela Superior de Ingeniería y Tecnología, Universidad Internacional de La Rioja, Avenida de La Paz 137, 26002, Logroño, Spain.

1 Introduction

The calculation of the solution of a nonlinear equation $f(x) = 0$, where $f : D \subset \mathbb{R} \longrightarrow \mathbb{R}$, is required in many scientific and engineering processes. Let us denote this solution by x^*. Since the resolution of these problems is often difficult and there exist few analytical methods to solve them, we use iterative algorithms to approximate the solution. Starting with an initial approximation of the solution, iterative methods generate a sequence of points that under certain criteria converge to x^*. The general expression of this iterative process is

$$x_{k+1} = g(x_k), \qquad k \geq 0,$$

where g is the fixed point function that defines the method, and $x_0 \in \mathbb{R}$ is the initial estimation. However, if more than one previous iterate are used to obtain the following approximation to x^*, we classify the iterative scheme as a method with memory, being its general expression

$$x_{k+1} = g(x_{k-m}, \ldots, x_{k-1}, x_k), \qquad k \geq m,$$

and requires the starting points $x_0, x_1, \ldots, x_m \in \mathbb{R}$.
In the current decades, many authors have devoted their research to the design and analysis of new iterative methods. In [1–3] we can find extensive studies on such schemes, taking into account their order of convergence p, and their efficiency in terms of the number of different functional evaluations d performed in each iteration of the method. Kung and Traub conjectured in [1] that a method without memory has at most order 2^{d-1}. When this upper bound is reached, the method is classified as an optimal iterative process. However, the order of convergence of methods with memory is not limited by this value. Therefore, it is common to include more than one previous iteration in order to design iterative schemes with higher order of convergence, resulting in methods with memory. In this paper, starting from a family of iterative methods of order two, we use this technique to design an iterative scheme that improves its order of convergence without including more functional evaluations.

[1] neus.garrido@unir.net

2 Introduction of memory and convergence analysis

The starting point of this study is a family of iterative methods presented in [4], whose iterative expression is

$$x_{k+1} = x_k - H(t_k), \qquad k = 0, 1, 2, \ldots, \tag{1}$$

being $H(t)$ a weight function of variable $t_k = \frac{f(x_k)}{f'(x_k)}$. It is also proved in [4] that family (1) converges quadratically when the weight function holds $H(0) = 0$, $H'(0) = 1$ and $|H''(0)| < \infty$. In addition, in [5] the authors select a family of iterative methods that belongs to (1) with expression

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} - \frac{\alpha}{2}\left(\frac{f(x_k)}{f'(x_k)}\right)^2, \qquad k = 0, 1, 2, \ldots \tag{2}$$

Let us note that (2) is obtained using in (1) the weight function $H(t) = t + \alpha\frac{t^2}{2}$, $\alpha \in \mathbb{R}$. The class (2) has quadratic convergence for any α and its error equation is

$$e_{k+1} = \left(c_2 - \frac{\alpha}{2}\right) e_k^2 + \mathcal{O}(e_k^3),$$

where $e_k = x_k - x^*$, $\forall k$, and $c_2 = \frac{f''(x^*)}{2f'(x^*)}$. From the lower term in the error equation, the order of convergence of the family can increase a unit when $\alpha = 2c_2$. Since x^* is unknown, different approximations for $f'(x^*)$ and $f''(x^*)$ are used in [5] obtaining methods with memory with order $p = 1 + \sqrt{2} \approx 2.4142$. Mainly, these approximations use quadratic polynomials and rational funtions.
Hereinafter, we are analyzing an approximation for $f''(x^*)$ using high-order degree polynomials in order to achieve an even greater increase in the order of convergence. In particular, we are going to approximate $f''(x^*)$ using cubic interpolation polynomials. Let us consider the general expression of a cubic polynomial

$$p(x) = a + bx + cx^2 + dx^3.$$

Coefficients a, b, c and d are obtained imposing the following conditions:

$$p(x_k) = f(x_k), \quad p(x_{k-1}) = f(x_{k-1}), \quad p'(x_k) = f'(x_k), \quad p'(x_{k-1}) = f'(x_{k-1}).$$

Then, we can approximate $f'(x^*) \approx f'(x_k)$ and $f''(x^*) \approx p''(x_k)$, so the approximation for parameter α is given by:

$$\alpha_k = \frac{p''(x_k)}{f'(x_k)} = \frac{-6f(x_k) + 6f(x_{k-1}) + 2(x_k - x_{k-1})(2f'(x_k) + f'(x_{k-1}))}{f'(x_k)(x_k - x_{k-1})^2}. \tag{3}$$

We denote the iterative scheme obtained after replacing in (2) the parameter α_k defined in (3) as CS method. Theorem 1 shows the improvement in the quadratic order of convergence if approximation (3) is considered.

Theorem 1. *Let $f : I \subset \mathbb{R} \longrightarrow \mathbb{R}$ be a sufficiently differentiable function in an open interval I. If $x^* \in I$ is a simple root of $f(x) = 0$ and x_0 and x_1 are initial estimations close to enough to x^*, then the iterative method CS converges to x^* with order of convergence $p = 1 + \sqrt{3} \approx 2.7321$.*

Therefore, the inclusion of memory by means of cubic interpolation polynomials makes it possible to increase the order of convergence of family (2) without increasing the number of different functional evaluations.

3 Stability analysis: basins of attraction

In Section 2, the order of convergence of the CS method has been introduced. Moreover, the analysis of the stability of the iterative scheme in terms of the initial estimations using a complex dynamical study [6] is also useful. Since CS is a method with memory, we must use tools from multidimensional real dynamics [7] to carry out this analysis .
Let us note that CS scheme is a method with memory with general expression $x_{k+1} = g(x_{k-1}, x_k)$. In order to calculate its fixed points, the authors in [7] define an auxiliar vectorial function $G : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ and then the discrete dynamical system

$$G(z, x) = (x, g(z, x)),$$

where we denote $z = x_{k-1}$ and $x = x_k$. Then, the orbit of a point $(z, x) \in \mathbb{R}^2$ is the set of its successive images by G, i.e. $\{(z, x), G(z, x), G^2(z, x), \ldots, G^m(z, x), \ldots\}$.
We consider that the iterative scheme under study is applied on a nonlinear function $f(x)$. Thus, the fixed points of the associated vectorial function G satisfy $G(z, x) = (z, x)$. In addition, when they are different to the roots of $f(x)$ they are called strange fixed points. The asymptotical behaviour of the fixed points (z^F, x^F) of G is classified depending on the eigenvalues λ_1 and λ_2 of the Jacobian matrix $G'(z^F, x^F)$. According to Robinson [8], a fixed point is attracting when $|\lambda_{1,2}| < 1$, repelling if $|\lambda_{1,2}| > 1$ and (z^F, x^F) is called saddle point if $|\lambda_1| > 1$ but $|\lambda_2| < 1$.
For an attracting fixed point x^*, its basin of attraction is defined as the set of preimages of any order that converge to it, that is

$$\mathcal{A}(x^*) = \{(z_0, x_0) \in \mathbb{R}^2 : G^m(z_0, x_0) \longrightarrow x^*, m \to \infty\}.$$

We can represent the basins of attraction of the roots of a given nonlinear function $f(x)$ using the dynamical planes. In this plot, the real plane is divided into a mesh of points that are taken as initial estimates to iterate the iterative method used to approximate the roots of $f(x)$. After successively applying the asociated vectorial operator G to each point in the plane, if it converges to any of the roots, it is represented in the corresponding colour and in black otherwise. In this way, we are able to determine the stability of the iterative method in terms of the set of initial estimations that converge to the roots of the nonlinear function.
Therefore, dynamical planes are useful for selecting the best initial estimations required to apply an iterative method to approximate the solution of a nonlinear equation. In Section 4 we test the performance of CS method by solving different nonlinear functions. Previously, we apply CS method to the considered equations in order to compare its stability over different examples and also with the classical Secant's method, whose iterative expression is

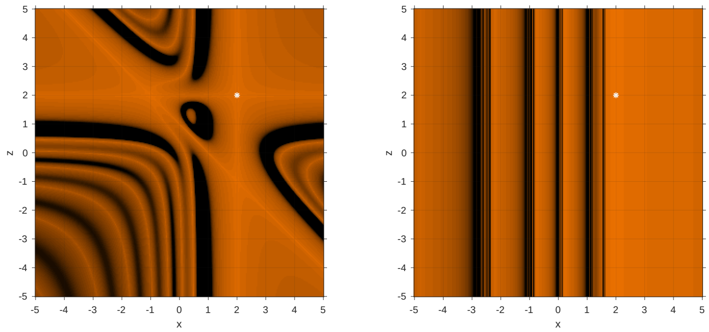
$$x_{k+1} = x_k - \frac{(x_k - x_{k-1})f(x_k)}{f(x_k) - f(x_{k-1})}, \qquad k = 1, 2, \ldots$$

Let us note that Secant's method is a scheme with memory with order of convergence $p \approx 2.41$. Figures 1, 2 and 3 show the dynamical planes of the nonlinear funcions in Table 1, respectively.

Function	Roots
$f_1(x) = (x-1)^3 - 1$	$x^* = 2$
$f_2(x) = x^2 - e^x - 3x + 2$	$x^* \approx 0.257530.$
$f_3(x) = \sin(x) - x^2 + 1$	$x_1^* \approx 1.409624$, $x_2^* \approx -0.636732$

Table 1: Test nonlinear functions for the stability and the numerical analysis

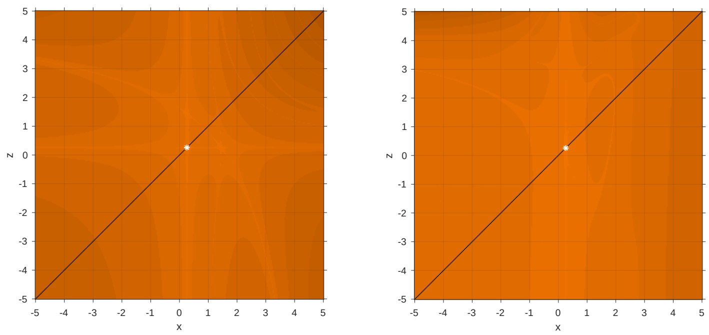
The real roots of the nonlinear functions, denoted by x^*, are also fixed points of the vectorial fixed point function associated to CS method applied to them. They are represented with white stars in the dynamical planes and their basin of attraction with colour orange for the roots of $f_1(x)$ and $f_2(x)$. We have depicted in orange and blue the initial estimations that belong to the basins of attraction of the two real roots of $f_3(x)$. The convergence in the dynamical planes is set when the difference between a point of the orbit of each initial guess and a root of the considered function is lower than 10^{-5} with a maximum of 50 iterations of the method. As CS is a method with memory, the axes in the real plane represent the current and the previous iterations, x and z, respectively.
Figure 1 shows the dynamical planes corresponding to $f_1(x)$. For any initial guess, the most of the points converge to x^*. However, there are points in black that do not belong to its basin of attraction, with wider regions represented in black for Secant's method.



(a) Secant (b) CS

Figure 1: Basins of attraction for $f_1(x)$

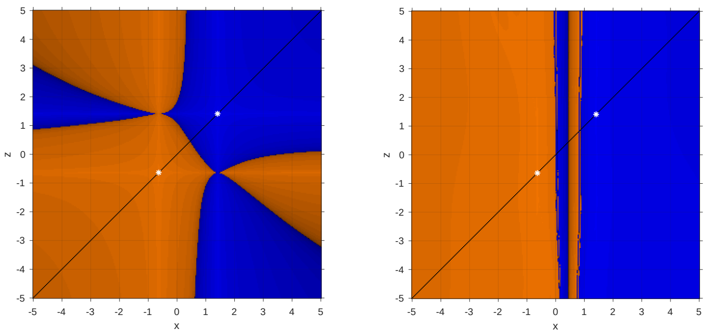
Figures 2 and 3 show that every initial guess converge to a root of the nonlinear equation. This fact shows the stability of both Secant and CS methods. Moreover, the colour intensity denotes that the initial estimate requires more iterations until its orbit converges to a root.



(a) Secant (b) CS

Figure 2: Basins of attraction for $f_2(x)$

Therefore, the dynamical planes previously shown (Figures 1-3) highlight the stability of method CS depending on the initial estimate considered for these nonlinear test functions, even improving the stability of Secant's iterative scheme.



(a) Secant (b) CS

Figure 3: Basins of attraction for $f_3(x)$

4 Numerical experiments

In this section we will approximate the roots of the nonlinear test functions $f_{1-3}(x)$ in Table 1 using Secant's method and CS iterative scheme. Based on Figures 1-3, we will take as an initial estimation for solving the equations different points in the basins of attraction denoted in orange or blue. This fact will guarantee the convergence to a root of the corresponding function. Furthermore, we have choose for simplicity $z_0 = x_0 + 0.1$ in all the cases.
The numerical experiments have been carried using software Matlab R2018b. The convergence is set when $|x_{k+1} - x_k| < 10^{-5}$ or $|f(x_{k+1})| < 10^{-5}$, being the number of iterations lower than 50. Table 1 shows, for each nonlinear function, the initial estimation x_0, the number of iterations required to converge to the root, the approximation of x^*, the difference between the two last iterations, the value of the funtion in the last iteration and the Approximated Computational Order of Convergence (ACOC) defined in [9] as

$$p \approx ACOC = \frac{\ln(|x_{k+1} - x_k|/|x_k - x_{k-1}|)}{\ln(|x_k - x_{k-1}|/|x_{k-1} - x_{k-2}|)}, \qquad k = 2, 3, \ldots$$

Table 1 gathers the performance of the numerical experiments. Let us note that the numerical performance of both methods is acceptable, since in all cases the initial estimates converge to the root of each function. Moreover, method CS requires in general less number of iterations than Secant's scheme to approximate the solution of the equation more accurately.

5 Conclusions

Starting from a family of iterative schemes with quadratic convergence, a method with memory with order of convergence 2.7321 has been designed. This method has been obtained using a third-degree interpolation polynomial for the approximation of the accelerating parameter that is present in the initial family. In adddition, a stability analysis depending on the initial estimations has been performed for different nonlinear test functions, showing wide basins of attraction corresponding to the roots of the functions. Finally, the numerical performance of the proposed class has been compared with Secant's scheme, obtaining accurate approximations of the roots of the considered test functions.

f	$\mathbf{x_0}$	**Method**	**iter**	$\mathbf{x^*}$	$\lvert\mathbf{x_{k+1}-x_k}\rvert$	$\lvert\mathbf{f(x_{k+1})}\rvert$	**ACOC**
$f_1(x)$	1.2	Secant	18	2	3.8278e–06	5.1595e–09	1.6013
		CS	16	2	5.2281e–08	7.1450e–22	2.9175
	-0.5	Secant	14	2	2.1585e–07	4.9458e–11	1.6407
		CS	7	2	3.1737e–07	1.5984e–19	2.8926
	-4	Secant	23	2	6.9851e–06	1.3593e–08	1.5674
		CS	10	2	1.0133e–11	5.2026e–33	2.9760
$f_2(x)$	0	Secant	4	0.2575302854	5.0514e–08	2.1496e–12	2.1675
		CS	3	0.2575302854	7.0601e–11	2.2390e–29	2.2410
	1	Secant	4	0.2575302855	1.0718e–06	2.2724e–10	1.8087
		CS	4	0.2575302854	9.2439e–15	1.9607e–39	2.8431
	2	Secant	6	0.2575302854	1.2809e–08	1.7620e–13	1.7770
		CS	4	0.2575302854	1.1639e–14	1.5604e–38	3.5723
$f_3(x)$	1	Secant	6	1.409624004	3.8472e–08	2.0162e–12	1.6896
		CS	4	1.409624004	3.4818e–08	7.2788e–22	3.0893
	-2	Secant	6	-0.6367326508	1.6286e–07	1.1554e–11	1.6747
		CS	4	-0.6367326508	3.6851e–11	5.5928e–30	2.9962
	0.25	Secant	8	-0.6367326508	1.5058e–07	9.7093e–12	1.6260
		CS	6	1.409624004	1.1850e–07	2.1495e-20	2.9196

Table 2: Numerical results for $f_1(x)$, $f_2(x)$ and $f_3(x)$

Acknowledgements

The authors were supported by the internal research project ADMIREN of Universidad Internacional de La Rioja (UNIR). The first author was also partially supported by PGC2018-095896-B-C22 (MCIU/AEI/FEDER, UE).

References

[1] Kung, H.T., Traub, J.F. Optimal order of one-point and multipoint iteration. *Journal of the Association for Computing Machinery*, 21:643–651, 1974.

[2] Amat, S., Busquier, S. Advances in iterative methods for nonlinear equations. Springer, 2016.

[3] Petković, M.S., Neta, B., Petković, L.D., Džunic, J. Multipoint methods for solving nonlinear equations. Elsevier, 2013.

[4] Chicharro, F.I., Cordero, A., Garrido, N., Torregrosa, J.R. Generating root-finder iterative methods of second order: Convergence and stability. *Axioms*, 8(2):55, 2019.

[5] Chicharro, F.I., Garrido, N., Sarría, I., Orcos, L. Different approximations of the parameter for low-order iterative methods with memory. Proceedings of the XXVI Congreso de Ecuaciones Diferenciales y Aplicaciones. XVI Congreso de Matemática Aplicada. Universidad de Oviedo, Servicio de Publicaciones, 2021, pp. 130–134.

[6] Devaney, R.L. An Introduction to Chaotic Dynamical Systems. Addison-Wesley, 1964.

[7] Campos, B., Cordero, A., Torregrosa, J.R., Vindel, P. A multidimensional dynamical approach to iterative methods with memory. *Applied Mathematics and Computation*, 271:701–715, 2015.

[8] Robinson, R.C. An Introduction to Dynamical Systems: Continuous and Discrete. American Mathematical Society, 2012.

[9] Cordero, A., Torregrosa, J.R. Variants of Newton's method using fifth-order quadrature formulas. *Applied Mathematics and Computation*, 190:686–698, 2007.