Applied Mathematics Letters 23 (2010) 472–478

Contents lists available at ScienceDirect

Applied Mathematics Letters

journal homepage: www.elsevier.com/locate/aml

On some computational orders of convergence

Miquel Grau-Sánchez [a,*], Miquel Noguera [a], José Manuel Gutiérrez [b]

[a] Technical University of Catalonia, Department of Applied Mathematics II, Jordi Girona 1-3, Omega, 08034 Barcelona, Spain

[b] University of La Rioja, Department of Mathematics and Computation, Luis Ulloa s/n, 26004 Logroño, Spain

A R T I C L E I N F O

Article history:
Received 20 July 2009
Received in revised form 17 November 2009
Accepted 4 December 2009

Keywords:
Order of convergence
Nonlinear equations
Iterative methods

A B S T R A C T

Two variants of the Computational Order of Convergence (COC) of an iterative method for solving nonlinear equations are presented. Furthermore, the way to approximate the COC and the new variants to the local order of convergence is analyzed. The new definitions given here does not involve the unknown root. Numerical experiments using adaptive arithmetic with multiple precision and a stopping criteria are implemented without using any known root.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

Iterative methods for solving a nonlinear equation $f(x) = 0$, where $f : D \subseteq \mathbb{R} \to \mathbb{R}$, usually consider a sequence $\{x_n\}$, defined by

$$x_{n+1} = \phi(x_n), \quad n \geq 0, \tag{1}$$

where ϕ is the iteration function. A sequence $\{x_n\}$ is said to converge to α with *order of convergence* $\rho \in \mathbf{R}$, $\rho \geq 1$, if there exists a constant $C \in \mathbf{R} \setminus \{0\}$ such that

$$\lim_{n\to\infty} \frac{e_{n+1}}{e_n^{\rho}} = C,$$

where $e_n = x_n - \alpha$ is the error in the nth iterate. For one-step methods, like the one given in (1), the error equation is:

$$e_{n+1} = Ce_n^{\rho} + De_n^{\rho+1} + \cdots, \tag{2}$$

where C and D are real numbers. The nonzero constant C is said to be the asymptotic error. The order of convergence of an iterative method is the order of the corresponding sequence. If it is ρ, then the method approximately multiplies by ρ the number of correct decimals after each iteration.
Next, we give the definitions of Computational Order of Convergence (COC) [1, 2000], Approximated Computational Order of Convergence (ACOC) and Extrapolated Computational Order of Convergence (ECOC).

Definition 1 (*Computational Order of Convergence, COC*). The computational order of convergence (COC) of a sequence $\{x_n\}_{n\geq 0}$ is defined by

$$\bar{\rho}_n = \frac{\ln |e_{n+1}/e_n|}{\ln |e_n/e_{n-1}|}, \tag{3}$$

where x_{n-1}, x_n and x_{n+1} are three consecutive iterations near the root α and $e_n = x_n - \alpha$.

* Corresponding author.
E-mail addresses: miquel.grau@upc.edu (M. Grau-Sánchez), miquel.noguera@upc.edu (M. Noguera), jmguti@unirioja.es (J.M. Gutiérrez).

0893-9659/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.aml.2009.12.006

After the work of Weerakoon and Fernando [1], many other authors have considered the COC in their research. In [2–16] the computation of COC is used in each of them. In all these papers the COC is used to test numerically the order of convergence of the methods previously presented which order have been theoretically studied. One of the main drawback of the COC is that it involves the exact root α, which in a real situation it is not known a priori. To avoid this, we introduce a variant of COC, that does not use the exact root. Previously for several variables this concept was considered in [17,18].

Definition 2 (*Approximated Computational Order of Convergence, ACOC*). The approximated computational order of convergence (ACOC) of a sequence $\{x_n\}_{n\geq 0}$ is defined by

$$\hat{\rho}_n = \frac{\ln\left|\hat{e}_{n+1}/\hat{e}_n\right|}{\ln\left|\hat{e}_n/\hat{e}_{n-1}\right|}, \tag{4}$$

where $\hat{e}_n = x_n - x_{n-1}$.

Like in the ACOC to avoid formulae involving the exact root α, we begin with three consecutive iterates $x_n,\ x_{n-1},\ x_{n-2}$, and using Aitken's extrapolation procedure [19] we give the following approximation of α

$$\tilde{\alpha}_n = x_n - \frac{(\Delta x_{n-1})^2}{\Delta^2 x_{n-2}}, \quad n \geq 2, \tag{5}$$

where Δ is the forward difference operator, $\Delta x_k = x_{k+1} - x_k$. Then, we can define a new approximation for the error $\tilde{e}_n = x_n - \tilde{\alpha}_n$ and a new computational order of convergence:

Definition 3 (*Extrapolated Computational Order of Convergence, ECOC*). The extrapolated computational order of convergence (ECOC) of a sequence $\{x_n\}_{n\geq 0}$ is defined by

$$\tilde{\rho}_n = \frac{\ln\left|\tilde{e}_{n+1}/\tilde{e}_n\right|}{\ln\left|\tilde{e}_n/\tilde{e}_{n-1}\right|}, \tag{6}$$

where $\tilde{e}_n = x_n - \tilde{\alpha}_n$ and $\tilde{\alpha}_n$ is given by (5).

As we show later, for all sequence $\{x_n\}$ converging to α, with a starting point x_0 close enough to α, the values of $\bar{\rho}_n$, $\hat{\rho}_n$ and $\tilde{\rho}_n$ converge, when $n \to \infty$, to ρ.
In numerical problems where a huge number of significant digits of the solution is needed it is required the use of methods with a high order of convergence together with adequate arithmetics. There are different libraries in Fortran [20] or C [21] working with a multiple precision arithmetic or symbolic manipulators, as Maple, that allow to work with an adaptive arithmetic, that is to update the length of the mantissa at each step by means of the formula

$$\texttt{Digits} := [\rho \times (-\log|e_n| + j)], \tag{7}$$

where ρ is the order of convergence of the method and $[x]$ denotes the integer part of x. Notice that the length of the mantissa is increased approximately by the order of convergence ρ. We have numerically checked the value of j, by varying it between 1 and 5, in order to have enough accuracy in the computation of the iterates $\{x_n\}_{n\geq 0}$. We have realized that the minimum value that guarantees all the significant digits required is $j = 2$. Consequently, hereof we consider $j = 2$ in formula (7).
In addition, to compute e_n, $\hat{e}_n$ or $\tilde{e}_n$ with an appropriate number of figures, using Definitions 1, 2 or 3 we must to enlarge the mantissa in the computation of $x_{n+1}, x_n, x_{n-1}, \ldots$ with at least four additional significant digits.

2. Computational Order of Convergence (COC)

A relationship between $\bar{\rho}_n$ and ρ is derived. In fact, we prove that $\bar{\rho}_n$ converges to ρ when $e_{n-1} \to 0$. That is $\bar{\rho}_n \approx \rho$, in the sense that $\lim_{n\to\infty} \frac{\bar{\rho}_n}{\rho} = 1$.

Proposition 2.1. *If $\bar{\rho}_n$ is the COC defined in (3) and ρ is the order of convergence, then*

$$\bar{\rho}_n \approx \rho\left|1 - \frac{N_n}{\rho(\ln M_n + N_n)}\right|, \quad \text{where } M_n = \left|C\, e_{n-1}^{\rho-1}\right| \text{ and } N_n = \left|\frac{D}{C} e_{n-1}\right| \tag{8}$$

and C and D are given in (2).

Proof. To prove (8) we express $\bar{\rho}_n$ in terms of e_{n-1}. We denote $O_q = O(e_{n-1}^q)$, and taking into account $e_{n+1} = C^{\rho+1} e_{n-1}^{\rho^2} + \rho\, C^{\rho} D\, e_{n-1}^{\rho^2+1} + O_{\rho^2+2}$, we have

$$\begin{aligned}\frac{e_{n+1}}{e_n} &= \frac{C^\rho e_{n-1}^{\rho^2-\rho} + \rho C^{\rho-1} D\, e_{n-1}^{\rho^2-\rho+1} + O_{\rho^2-\rho+2}}{1+\frac{D}{C} e_{n-1} + O_2}\\ &= C^\rho e_{n-1}^{\rho^2-\rho} + (\rho-1)\, C^{\rho-1} D\, e_{n-1}^{\rho^2-\rho+1} + O_{\rho^2-\rho+2},\end{aligned} \tag{9}$$

and

$$\frac{e_n}{e_{n-1}} = C e_{n-1}^{\rho-1} + D\, e_{n-1}^{\rho} + O_{\rho+1}. \tag{10}$$

From (9), (10) and using the equivalence $\ln(1+x) \approx x$, for x closer enough to zero, yields

$$\bar{\rho}_n = \frac{\ln|e_{n+1}/e_n|}{\ln|\,e_n/e_{n-1}|} \approx \rho \left| \frac{\ln M_n + \frac{\rho-1}{\rho} N_n}{\ln M_n + N_n} \right| \approx \rho \left| 1 - \frac{N_n}{\rho(\ln M_n + N_n)} \right|,$$

where M_n and N_n are defined in (8). □

Notice that for the calculus of the COC (3) and for updating the adaptive arithmetic process (7) it is necessary to know the exact root α. In this case the following stopping criteria is applied:

$$|e_n| = |x_n - \alpha| < 0.5 \cdot 10^{-\eta}, \tag{11}$$

where η is the number of correct decimals and $0.5 \cdot 10^{-\eta}$ is the required accuracy.

3. Approximated Computational Order of Convergence (ACOC)

A relationship between $\hat{\rho}_n$, and ρ is obtained. A new technique to update the number of significant digits in an adaptive multi-precision arithmetic is given and a new stopping criteria is suggested.

Proposition 3.1. *If we set $e_n = x_n - \alpha$ and $\hat{e}_n = x_n - x_{n-1}$, then*

$$e_n \approx C^{1/(1-\rho)} \left(\frac{\hat{e}_n}{\hat{e}_{n-1}} \right)^{\rho^2/(\rho-1)}, \tag{12}$$

where ρ be the order of convergence and C is given in (2).

Proof. Now, we write $\hat{e}_n/\hat{e}_{n-1}$ in terms of e_{n-2}:

$$\begin{aligned}\frac{\hat{e}_n}{\hat{e}_{n-1}} = \frac{e_n - e_{n-1}}{e_{n-1} - e_{n-2}} &= \frac{C^{\rho+1} e_{n-2}^{\rho^2} - C\, e_{n-2}^{\rho} + O\left(e_{n-2}^{\rho+1}\right)}{C\, e_{n-2}^{\rho} + O\left(e_{n-2}^{\rho+1}\right) - e_{n-2}}\\ &= C\, e_{n-2}^{\rho-1} \left(1 + O\left(e_{n-2}\right)\right).\end{aligned} \tag{13}$$

Putting

$$e_{n-2} = C^{-(\rho+1)/\rho^2} e_n^{1/\rho^2} \left(1 + O\left(e_n^{1/\rho^2}\right)\right), \tag{14}$$

and substituting (14) in (13) we have

$$\frac{\hat{e}_n}{\hat{e}_{n-1}} = C^{1/\rho^2} e_n^{(\rho-1)/\rho^2} \left(1 + O\left(e_n^{1/\rho^2}\right)\right), \tag{15}$$

that can be expressed by $e_n^{(\rho-1)/\rho^2} \approx C^{-1/\rho^2} \frac{\hat{e}_n}{\hat{e}_{n-1}}$, and the proof is completed. □

The result given in (12) allows us to substitute the error in (7) by a expression that does not involve the exact root. Indeed, we implement the following adaptive multi-precision arithmetic scheme:

$$\mathtt{Digits} := \left[\frac{\rho^3}{\rho-1} \times \left(-\log \left| \frac{\hat{e}_n}{\hat{e}_{n-1}} \right| + 2 \right) \right]. \tag{16}$$

Moreover, from (12) we propose the following stopping criteria, instead of (11):

$$\left| \frac{\hat{e}_n}{\hat{e}_{n-1}} \right| < 0.5 \cdot 10^{-\eta(\rho-1)/\rho^2}. \tag{17}$$

Next result shows the relationship between the ACOC and the order of convergence of a sequence (1).

Proposition 3.2. *If $\hat{\rho}_n$ is the ACOC defined in (4) and ρ is the order of convergence, then*

$$\hat{\rho}_n \approx \rho \left| 1 - \frac{N_{n-1}}{\rho(\ln M_{n-1} + N_{n-1})} \right|, \quad \text{where } M_{n-1} = \left| C\, e_{n-2}^{\rho-1} \right| \text{ and } N_{n-1} = \left| \frac{D}{C} e_{n-2} \right| \tag{18}$$

and C and D are introduced in (2).

Proof. As in the proof of the previous proposition, we have

$$\frac{\hat{e}_n}{\hat{e}_{n-1}} = \frac{e_n - e_{n-1}}{e_{n-1} - e_{n-2}} = \frac{C^{\rho+1}\, e_{n-2}^{\rho^2} + \rho C^{\rho} D\, e_{n-2}^{\rho^2+1} + \cdots - C\, e_{n-2}^{\rho} - D\, e_{n-2}^{\rho+1} + O\left(e_{n-2}^{\rho+2}\right)}{C\, e_{n-2}^{\rho} + D\, e_{n-2}^{\rho+1} + O\left(e_{n-2}^{\rho+2}\right) - e_{n-2}}$$

$$= C\, e_{n-2}^{\rho-1} + D\, e_{n-2}^{\rho} + O\left(e_{n-2}^{\rho+1}\right). \tag{19}$$

In a similar way,

$$\frac{\hat{e}_{n+1}}{\hat{e}_n} = C^{\rho}\, e_{n-2}^{\rho(\rho-1)} + (\rho-1) C^{\rho-1} D e_{n-2}^{\rho^2\rho+1} + O\left(e_{n-2}^{\rho^2-\rho+2}\right). \tag{20}$$

From (19) and (20) we have

$$\hat{\rho}_n = \frac{\ln\left|\hat{e}_{n+1}/\hat{e}_n\right|}{\ln\left|\hat{e}_n/\hat{e}_{n-1}\right|} \approx \rho \left| \frac{\ln M_{n-1} + \frac{\rho-1}{\rho} N_{n-1}}{\ln M_{n-1} + N_{n-1}} \right| \approx \rho \left| 1 - \frac{N_{n-1}}{\rho(\ln M_{n-1} + N_{n-1})} \right|,$$

with M_{n-1} and N_{n-1} defined in (18). This completes the proof. □

4. Extrapolated Computational Order of Convergence (ECOC)

A relationship between $\tilde{\rho}_n$ and ρ, a new technique to update the number of significant digits in an adaptive multi-precision arithmetic and a new stopping criteria are given.

Proposition 4.1. *If we put $e_n = x_n - \alpha$ and $\tilde{e}_n = x_n - \tilde{\alpha}_n$, then*

$$e_n \approx C^{\sigma}\, \tilde{e}_n^{\rho^2/(2\rho-1)}, \quad \text{where } \sigma = \frac{\rho-1}{2\rho-1}. \tag{21}$$

Proof. Taking into account $e_{n-2} = C^{-1/\rho}\, e_{n-1}^{1/\rho}(1 + O(e_{n-1}^{1/\rho}))$, we write $\tilde{e}_n$ in terms of e_{n-1}:

$$\tilde{e}_n = \frac{(e_n - e_{n-1})^2}{e_n - 2e_{n-1} + e_{n-2}} = \frac{C^2\, e_{n-1}^{2\rho} - 2C\, e_{n-1}^{\rho+1} + e_{n-1}^2 + O(e_{n-1}^{\rho+2})}{C\, e_{n-1}^{\rho} - 2e_{n-1} + C^{-1/\rho} e_{n-1}^{1/\rho}\left(1 + O(e_{n-1}^{1/\rho})\right)}$$

$$= C^{1/\rho}\, e_{n-1}^{(2\rho-1)/\rho}\left(1 + O(e_{n-1}^{2/\rho})\right). \tag{22}$$

Now, from (22) and $e_{n-1} = C^{-1/\rho}\, e_n^{1/\rho}(1 + O(e_n^{1/\rho}))$, we get

$$\tilde{e}_n = C^{1/\rho}\left[C^{-1/\rho}\, e_n^{1/\rho}\left(1 + O\left(e_n^{1/\rho}\right)\right)\right]^{(2\rho-1)/\rho} \cdot \left[1 + O\left(\left\{C^{-1/\rho}\, e_n^{1/\rho}\left(1 + O\left(e_n^{1/\rho}\right)\right)\right\}^{2/\rho}\right)\right]$$

$$= C^{(1-\rho)/\rho^2} e_n^{(2\rho-1)/\rho^2}\left(1 + O\left(e_n^{(2\rho-1)/\rho^2}\right)\right). \tag{23}$$

From (23), we have $e_n^{(2\rho-1)/\rho^2} \approx C^{(\rho-1)/\rho^2}\tilde{e}_n$ from which the proof immediately follows. □

Notice that (21) allow us to implement an iterative method (1) with a multi-precision adaptive arithmetic. We consider instead of (7) the expression:

$$\texttt{Digits} := \left[\frac{\rho^3}{2\rho-1} \times \left(-\log|\tilde{e}_n| + 2\right)\right]. \tag{24}$$

In addition, as an alternative to (11), (21) provides the following stopping criteria

$$|\tilde{e}_n| < 0.5 \cdot 10^{-\eta(2\rho-1)/\rho^2}. \tag{25}$$

The following result shows the relationship between the ECOC and the order of convergence.

Proposition 4.2. *If $\tilde{\rho}_n$ is the ECOC defined in (6) and $\rho \geq 2$ is the order of convergence, then*

$$\tilde{\rho}_n \approx \rho \left| 1 + \frac{(2\,\rho - 3)\, N_{n-1} + P_{n-1}}{\ln Q_{n-1}} \right|, \tag{26}$$

where $Q_{n-1} = |C^{2\rho-1}\, e_{n-2}^{(2\rho-1)\,(\rho-1)}|$, $N_{n-1} = \left|\frac{D}{C} e_{n-2}\right|$, $P_{n-1} = \rho |C^{-1} D^{\rho}\, e_{n-2} + 2\, C\, e_{n-2}^{\rho-1}|$, *and C and D are given in (2).*

Proof. Now, we write $\tilde{\rho}_n$in terms of e_{n-2}. To do that, we express $\tilde{e}_n$ and $\tilde{e}_{n+1}$ in terms of e_{n-2}.

$$\begin{aligned}\tilde{e}_n = \frac{(e_n - e_{n-1})^2}{e_n - 2e_{n-1} + e_{n-2}} &= \frac{C^2\, e_{n-2}^{2\rho} - 2C\, D\, e_{n-2}^{2\rho+1} + O\left(e_{n-2}^{2\rho+2}\right)}{e_{n-2} - 2\, C\, e_{n-2}^{\rho} + O\left(e_{n-2}^{\rho+1}\right)} \\ &= C^2\, e_{n-2}^{2\rho-1} + 2\, C\, D\, e_{n-2}^{2\rho} + O\left(e_{n-2}^{2\rho+1}\right).\end{aligned} \tag{27}$$

Next, by (27) we deduce

$$\begin{aligned}\tilde{e}_{n+1} &= C^2\, e_{n-1}^{2\rho-1} + 2CDe_{n-1}^{2\rho} + O\left(e_{n-1}^{2\rho+1}\right) \\ &= C^{2\rho+1}\, e_{n-2}^{(2\rho-1)\,\rho} + C^{2\rho}\, D\, e_{n-2}^{(2\rho-1)\,\rho+1} + O\left(e_{n-2}^{(2\rho-1)\,\rho+2}\right).\end{aligned} \tag{28}$$

From (27) and (28), we have

$$\frac{\tilde{e}_{n+1}}{\tilde{e}_n} = C^{2\rho-1} e_{n-2}^{(2\rho-1)\,(\rho-1)} + (2\rho - 3)\, C^{2\rho-2}\, D\, e_{n-2}^{(2\rho-1)\,(\rho-1)+1} + O\left(e_{n-2}^{(2\rho-1)\,(\rho-1)+2}\right). \tag{29}$$

Consequently,

$$\tilde{e}_{n-1} = \frac{(e_{n-1} - e_{n-2})^2}{e_{n-1} - 2e_{n-2} + e_{n-3}} = \frac{e_{n-2}^2 - 2\, C\, e_{n-2}^{\rho+1} + O\left(e_{n-2}^{\rho+1}\right)}{C^{-1/\rho}\, e_{n-2}^{1/\rho} - 1/\rho\, C^{-1-2/\rho}\, D\, e_{n-2}^{2/\rho} - 2\, e_{n-2} + O\left(e_{n-2}^{\rho}\right)},$$

and then

$$\tilde{e}_{n-1} = C^{1/\rho}\, e_{n-2}^{(2\rho-1)/\rho} + 1/\rho\, C^{-1}\, D\, e_{n-2}^2 + 2\, C^{2/\rho}\, e_{n-2}^{3-2/\rho} + \cdots. \tag{30}$$

From (28) and (31), we get

$$\frac{\tilde{e}_n}{\tilde{e}_{n-1}} = C^{(2\rho-1)/\rho} e_{n-2}^{(2\rho-1)\,(\rho-1)/\rho} - C^{(2\rho-2)/\rho}\, D\, e_{n-2}^{(2\rho-1)\,(\rho-1)/\rho+1/\rho} - 2\, C^2\, e_{n-2}^{2(\rho-1)} + O\left(e_{n-2}^{(2\rho^2-2\rho+1)/\rho}\right). \tag{31}$$

Finally, from (29) and (31), yields

$$\tilde{\rho}_n = \frac{\ln \left|\tilde{e}_{n+1}/\tilde{e}_n\right|}{\ln \left|\tilde{e}_n/\tilde{e}_{n-1}\right|} \approx \rho \left| \frac{\ln Q_{n-1} + (2\,\rho - 3) N_{n-1}}{\ln Q_{n-1} - P_{n-1}} \right| \approx \rho \left| 1 + \frac{(2\,\rho - 3) N_{n-1} + P_{n-1}}{\ln Q_{n-1}} \right|,$$

and the proof is complete. □

We point out that COC can be compute if $n \geq 1$, ACOC if $n \geq 2$ and ECOC if $n \geq 3$. If we have a method of higher order of convergence then multi-precision arithmetics is required and will be used in the case were necessary to obtain many correct figures.

5. Iterative methods and numerical results

We consider in this section three iterative methods, g_2, g_3 and g_4 with 2nd, 3rd and 4th local order of convergence respectively. These methods are called Newton's method, Chebyshev's method [13,22,23] and Schröder's method [24]. They are defined by:

$$g_2(x) = x - u(x), \tag{32}$$

$$g_3(x) = g_2(x) - \frac{1}{2} L(x) u(x), \tag{33}$$

$$g_4(x) = g_3(x) - \left(\frac{1}{2} L(x)^2 - M(x)\right) u(x), \tag{34}$$

Table 1
Test functions, their roots and their initial points.

Function	α	x_0
$f_1(x) = x^3 - 3x^2 + x - 2$	2.893289	2.5
$f_2(x) = x^3 + \cos x - 2$	1.172578	1.5
$f_3(x) = 2\sin x + 1 - x$	2.380061	2.5
$f_4(x) = (x+1)\,e^{-x} - 1$	0.557146	1.0
$f_5(x) = e^{x^2+7x-30} - 1$	3.0	2.94
$f_6(x) = e^{-x} + \cos(x)$	1.746140	1.5
$f_7(x) = x - 3\ln x$	1.857184	2.0

Table 2
Iteration number and higher bounds for $\bar{\rho}_{n-1}$, $\hat{\rho}_n$ and $\tilde{\rho}_n$.

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	$\Delta\bar{\rho}$	$\Delta\hat{\rho}$	$\Delta\tilde{\rho}$
g_2	13	12	11	12	13	11	12	$1.4 \cdot 10^{-5}$	$2.7 \cdot 10^{-23}$	$6.1 \cdot 10^{-9}$
g_3	9	8	7	8	9	7	8	$1.3 \cdot 10^{-7}$	$2.7 \cdot 10^{-34}$	$4.2 \cdot 10^{-12}$
g_4	7	7	6	7	7	6	6	$1.5 \cdot 10^{-9}$	$9.9 \cdot 10^{-50}$	$1.5 \cdot 10^{-16}$

where

$$u(x) = \frac{f(x)}{f'(x)}, \qquad L(x) = \frac{f''(x)}{f'(x)}u(x) \quad \text{and} \quad M(x) = \frac{f'''(x)}{3!\,f'(x)}u(x)^2.$$

We have tested the preceding methods on seven functions using the Maple computer algebra system. We have computed the root of each function for the same initial approximation x_0. Depending on the computational order of convergence used, COC (3), ACOC (4) or ECOC (6), the iterative method was stopped when the condition (11), (17) or (25) is fulfilled. Note that in all cases $\eta = 2200$.

The set of test functions presented here were previously considered in [25]. Table 1 shows the expression of these functions, the initial approximation, which is the same for all the methods and the root with seven significant digits.

Table 2 shows, for each method and function, the number of iterations needed to compute the root to the level of precision described. Note that independently of using (7), (16) or (24) the number of necessary iterations to get the desired precision is the same. In addition, the last three columns show an average for the error bounds produced in the computation of the corresponding Computational Orders of Convergence (COC, ACOC or ECOC). For instance, considering the COC and Newton's method g_2, let us denote $\Delta\bar{\rho}_k$ the error committed in the computation of the COC for each function f_k, $k = 1, \ldots, 7$. We calculate the average of these error bounds,

$$\Delta\bar{\rho} = \frac{1}{7}\sum_{k=1}^{7} |\Delta\bar{\rho}_k| = 1.4 \cdot 10^{-5}.$$

Then we can write the corresponding COC for Newton's method: $\bar{\rho}_{n-1} = \rho \pm \Delta\bar{\rho}$. The rest of the 8th column contains the average of the error bounds for the COC and methods g_3 and g_4. The 9th column shows the error bounds for the ACOC and each function g_j, $j = 2, 3, 4$ and the 10th column shows the error bounds for the ECOC and each function g_j, $j = 2, 3, 4$. Notice that we have $\hat{\rho}_n = \rho \pm \Delta\hat{\rho}$ for the ACOC and $\tilde{\rho}_n = \rho \pm \Delta\tilde{\rho}$ for the ECOC.

From these numerical tests, we can conclude that the ACOC produces the best approximations of the theoretical order of convergence of an iterative method. As we can see in Propositions 2.1 and 3.2, both COC and ACOC have the same asymptotic behavior, but ACOC has the advantage that it does not involve the expression of the root α we want to approximate that, in real problems, is not know in advance.

Acknowledgement

The work was partially supported by the grant Ref. MTM2008-01952/MTM, Spanish Ministry of Science and Innovation.

References

[1] S. Weerakoon, T.G.I. Fernando, A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 13 (2000) 87–93.
[2] M. Grau, M. Noguera, A variant of Cauchy's method with accelerated fifth-order convergence, Appl. Math. Lett. 17 (2004) 509–517.
[3] A.Y. Ozban, Some new variants of Newton's method, Appl. Math. Lett. 17 (2004) 677–682.
[4] C. Chun, Iterative methods improving Newton's method by the decomposition method, Comput. Math. Appl. 50 (2005) 1559–1568.
[5] G.H. Nedzhibov, V.I. Hasanov, M.G. Petkov, On some families of multipoint iteratie methods for solving nonlinear equations, Numer. Algorithms 42 (2006) 127–136.
[6] J. Kou, Y. Li, A variant of Chebyshev's method with sixth-order convergence, Numer. Algorithms 43 (2006) 273–278.
[7] C. Chun, Construction of Newton-like iteration methods for solving nonlinear equations, Numer. Math. 104 (2006) 297–315.
[8] C. Chun, A method for obtaining iterative formulas of order three, Appl. Math. Lett. 20 (2007) 1103–1109.
[9] J. Kou, Y. Li, X. Wang, Third-order modification of Newton's method, J. Comput. Appl. Math. 205 (2007) 1–5.

[10] C. Chun, Some third-order families of iterative methods for solving nonlinear equations, Appl. Math. Comput. 188 (2007) 924–933.
[11] C. Chun, Construction of third-order modifications of Newton's method, Appl. Math. Comput. 189 (2007) 662–668.
[12] C. Chun, On the construction of iterative methods with at least cubic convergence, Appl. Math. Comput. 189 (2007) 1384–1392.
[13] M. Grau-Sánchez, J.M. Peris, J.M. Gutiérrez, Accelerated iterative methods for finding solutions of a system of nonlinear equations, Appl. Math. Comput. 190 (2007) 1815–1823.
[14] T. Lukic, N.M. Ralevic, Geometric mean Newton's method for simple and multiple roots, Appl. Math. Lett. 21 (2008) 30–36.
[15] C. Chun, A simply constructed third-order modifications of Newton's method, J. Comput. Appl. Math. 219 (2008) 81–89.
[16] G.H. Nedzhibov, A family of multi-point iterative methods for solving systems of nonlinear equations, J. Comput. Appl. Math. 222 (2008) 244–250.
[17] A. Cordero, J.R. Torregrosa, Variants of Newton's method using fifth order quadrature formulas, Appl. Math. Comput. 190 (2007) 686–698.
[18] M.A. Noor, M. Waseem, Some iterative methods for solving a system of nonlinear equations, Comput. Math. Appl. 57 (2009) 101–106.
[19] A. Aitken, On Bernoulli's numerical solution of algebraic equations, Proc. Roy. Soc. Edinburgh 46 (1926) 289–305.
[20] D.H. Bailey, Multiprecision translation and execution of fortran programs, ACM Trans. Math. Software 19 (3) (1993) 288–319.
[21] L. Fousse, G. Hanrot, V. Lefèvre, P. Pélissier, P. Zimmermann, MPFR: A multiple -precision binary floating-point library with correct rounding, ACM Trans. Math. Software 33 (2) (2007) 1–15. Article 13.
[22] M. Grau, J.L. Díaz-Barrero, An improvement of the Euler–Chebyshev iterative method, J. Math. Anal. Appl. 315 (2006) 1–7.
[23] M. Grau-Sánchez, J.M. Gutiérrez, Some variants of the Chebyshev–Halley family with fifth-order of convergence, Int. J. Comput. Math. (2010), in press (doi:10.1080/00207160802208358).
[24] E. Schröder, Über unendlich viele Algorithmen zur Auflösung der Gleichungen, Math. Ann. 2 (1870) 317–365.
[25] M. Grau-Sánchez, Improvement of the efficiency of some three-step iterative like-Newton methods, Numer. Math. 107 (2007) 131–146.