Numer. Math. (2006) 104: 297–315
DOI 10.1007/s00211-006-0025-2

Numerische Mathematik

Changbum Chun

Construction of Newton-like iteration methods for solving nonlinear equations

Received: 17 January 2006 / Revised: 9 May 2006 / Published online: 8 August 2006

Abstract In this paper, we present a simple, and yet powerful and easily applicable scheme in constructing the Newton-like iteration formulae for the computation of the solutions of nonlinear equations. The new scheme is based on the homotopy analysis method applied to equations in general form equivalent to the nonlinear equations. It provides a tool to develop new Newton-like iteration methods or to improve the existing iteration methods which contains the well-known Newton iteration formula in logic; those all improve the Newton method. The orders of convergence and corresponding error equations of the obtained iteration formulae are derived analytically or with the help of Maple. Some numerical tests are given to support the theory developed in this paper.

Mathematics Subject Classification (2000) 65H05 · 65B99

1 Introduction

It is often necessary in scientific and engineering practices to find a root of a polynomial or an nonlinear equations. Probably the most well-known and widely used algorithm to find a root of such an equation is Newton's method that, when the root is simple, converges quadratically to it. However by such a method one can only find its solution near its initial approximation and approach its exact solution slowly. Improvements of the method, increasing the order of convergence, are usually obtained at the expense of an additional evaluation of the first or often higher derivative, additional evaluation of the function in the point of evaluation [32].

C. Chun
School of Liberal Arts & Education,
Korea University of Technology and Education,
Cheonan, ChungNam 330-708, Korea
E-mail: cbchun@kut.ac.kr

Recently, there has been some progress on Newton-like iteration methods improving Newton's method [1,2,8,9,11–14,17,31,33]. To obtain some of those iteration methods the Adomian decomposition method was applied in [1,8,9], He's homotopy perturbation method [2,11,14–16] and Liao's homotopy analysis method [20–29] by scientists and engineers because the latter two methods are to continuously deform a simple problem easy to solve into the difficult problem under study. The convergence of Newton-like methods are proved in, e.g., [7,19, 25]. However, the Newton-like methods developed so far are mostly based on a specific form of equations or systems that often lead to a restricted application to produce any further Newton-like formulae as the need arises.

In this paper the analytic approximate technique for nonlinear problems, namely the homotopy analysis method [22,24,26–28], which has already been successfully applied to many nonlinear problems, is employed to develop a numerical scheme that can be used in constructing new Newton-like iteration methods or further improving the already existing iterative methods to the order of convergence as high as one wants. To that end, the homotopy analysis method is applied to a transformed equation in general form equivalent to the nonlinear equation, not the nonlinear equation itself. It should be noted here that in this work the homotopy analysis method is applied to the nonlinear algebraic equations, not the differential equations.

The paper is organized as follows. The proposed scheme is described in Sect. 2 in detail together with some illustrations in Sect. 3 of various kinds of iteration formulae derived from the proposed scheme. We also give a detailed convergence analysis of the obtained iteration formulae analytically or with the help of symbolic computation of mathematical software package Maple. Lastly, numerical illustrations are given.

2 Scheme of constructing iterative methods

Consider the nonlinear equation

$$f(x) = 0. \tag{1}$$

Throughout the paper we assume that $f(x)$ has a simple root at α and γ is an initial guess close to α.

Let us transform the nonlinear equation (1) into the following canonical form:

$$x + N(x) = c \tag{2}$$

or

$$A(x) = L(x) + N(x) = c, \tag{3}$$

where c is a constant, L is a linear function and N a (nonlinear) function.

Using the embedding parameter $q \in [0, 1]$, we construct a family of equations called the zero-order deformation equation in homotopy analysis method [22–28]:

$$(1 - q)\,[L(v) - L(\xi)] = h\,q\,[A(v) - c], \tag{4}$$

where ξ is an initial guess close to α and h is a non-zero auxiliary parameter. Obviously, when $q = 0$ and $q = 1$, we have from (4) that

$$L(v) - L(\xi) = 0, \quad A(v) - c = 0. \tag{5}$$

So, as q increases from 0 to 1, (4) varies from $L(v)-L(\xi) = 0$ to $A(v)-c = 0$. In topology, this is called deformation, $L(v) - L(\xi)$ and $A(v) - c$ homotopic. Now, the solution v of (4) is a function of the embedding parameter q, denoting it by $v(q)$. Expanding $v(q)$ in a Taylor series gives

$$v(q) = v_0 + \sum_{m=1}^{+\infty} v_m q^m, \tag{6}$$

where

$$v_m = \frac{1}{m!} \frac{\partial^m v(q)}{\partial q^m}\bigg|_{q=0}. \tag{7}$$

The solution of (1) is, therefore, given by

$$v = \lim_{q\to 1} v(q) = v_0 + v_1 + v_3 + \cdots, \tag{8}$$

and its n-term approximant obtained by

$$v \approx \sum_{i=0}^{n-1} v_i. \tag{9}$$

Clearly, convergence of the series (8) depends on the auxiliary parameter h and the initial guess ξ [15,27]. As long as h and ξ are so properly chosen that the series (8) converges in a region, it converge to the exact solution in this region [25,27]. To obtain the approximate solution of (4), we expand $N(v)$ into Taylor series

$$N(v) = N(v_0)+N'(v_0)(qv_1+q^2v_2+\cdots)+\frac{1}{2!}N''(v_0)(qv_1+q^2v_2+\cdots)^2+\cdots. \tag{10}$$

Substituting (6) and (10) into the zero-order deformation equation (4), and the equating coefficients of like power of q, we obtain

$$q^0 : L(v_0) - L(\xi) = 0, \tag{11}$$

$$q^1 : L(v_1) - L(v_0) + L(\xi) = h\,[L(v_0) + N(v_0) - c], \tag{12}$$

$$q^2 : L(v_2) - L(v_1) = h\,[L(v_1) + N'(v_0)v_1], \tag{13}$$

$$q^3 : L(v_3) - L(v_2) = h\left[L(v_2) + N'(v_0)v_2 + \frac{1}{2!}N''(v_0)v_1^2\right], \tag{14}$$

$$q^4 : L(v_4)-L(v_3)=h\left[L(v_3) + N'(v_0)v_3 + N''(v_0)v_1v_2 + \frac{1}{3!}N'''(v_0)v_1^3\right] \tag{15}$$

$$\vdots$$

Equations (11) to (15) can be solved for the components of v only if L is invertible, so we assume that the inverse of the linear function L exists.
From (11), we can obtain one of its solutions

$$v_0 = \xi. \tag{16}$$

From equations (12) to (15), the components $v_1, v_2, v_3, \ldots$ of the solution of (1) can be recursively determined.

Some points should be emphasized here. The solution series given by the proposed scheme in the above contains the auxiliary parameter h, which provides us with a simple way to adjust convergence rate of solution series as will be seen in the next section. The solution series given by Adomian's decomposition method [3–5] is just a special case of the solution series given by the proposed scheme when $h = -1,\ \xi = c$. As a result, the iteration formulae proposed based on the Adomian decomposition method will be also derived from the proposed scheme [1,8–10].

3 Newton-like methods for equations of the form (2)

It is obvious that the nonlinear equation (1) can be rewritten in many different ways equivalent to (2). By way of illustration, the proposed scheme is in this section applied to four of those equivalent forms to argue that one can construct as many Newton-like iteration formulae improving the Newton method or existing iteration formulae as one likes. We also find order of convergence of most of the resulting iteration methods analytically or with the help of Maple.

As our first case of the form (2), we consider the following coupled system:

$$x + \frac{2[f(\gamma) + g(x)]}{f'(\gamma) + f'(x)} = \gamma, \tag{17}$$

$$g(x) = f(x) - f(\gamma) - \frac{f'(\gamma) + f'(x)}{2}(x - \gamma). \tag{18}$$

The system (17),(18) can be derived by rewriting the first equation of the following system equivalent to (1):

$$f(\gamma) + \frac{f'(\gamma) + f'(x)}{2}(x - \gamma) + g(x) = 0, \tag{19}$$

$$g(x) = f(x) - f(\gamma) - \frac{f'(\gamma) + f'(x)}{2}(x - \gamma). \tag{20}$$

It is easy to see that the method of Weerakoon and Fernando of order three [33] may be derived from (17) by letting $g(x) = 0$, solving for x,

$$x = \gamma - \frac{2f(\gamma)}{f'(\gamma) + f'(x)}, \tag{21}$$

and then replacing γ and x on the left-hand side of (21) with x_n and x_{n+1}, respectively, in combination with the approximation $f'(x) \approx f'(x_n - f(x_n)/f'(x_n))$. Applying (11) to (13) with $L(x) = x,\ N(x) = (2[f(\gamma) + g(x)])/(f'(\gamma) + f'(x))$, and $c = \gamma$, after some simplications, the components $v_0, v_1, v_2, v_3, \ldots$ are recursively determined as follows.

$$v_0 = \xi, \tag{22}$$

$$v_1 = h\frac{2f(\xi)}{f'(\gamma) + f'(\xi)}, \tag{23}$$

$$v_2 = 2h\ \frac{f(\xi)}{f'(\gamma)+f'(\xi)} + 4h^2\ \frac{f(\xi)f'(\xi)}{[f'(\gamma)+f'(\xi)]^2} - \ 4h^2\ \frac{f^2(\xi)f''(\xi)}{[f'(\gamma)+f'(\xi)]^3}, \tag{24}$$

and so on.
With $\xi = \gamma - \dfrac{f(\gamma)}{f'(\gamma)}$, we therefore have the first-order approximation

$$x \approx \gamma - \frac{f(\gamma)}{f'(\gamma)}, \tag{25}$$

and the second-order approximation

$$x \approx \gamma - \frac{f(\gamma)}{f'(\gamma)} + h\frac{2f(\xi)}{f'(\gamma) + f'(\xi)}, \tag{26}$$

and the third-order approximation

$$x \approx \gamma - \frac{f(\gamma)}{f'(\gamma)} + 4h\frac{f(\xi)}{f'(\gamma) + f'(\xi)} + 4h^2\frac{f(\xi)f'(\xi)}{[f'(\gamma) + f'(\xi)]^2} - 4h^2\ \frac{f^2(\xi)f''(\xi)}{[f'(\gamma) + f'(\xi)]^3} \tag{27}$$

and so on.
Writing $x_n = \gamma$ and $x_{n+1} = x$, we obtain the iteration methods

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{28}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + h\frac{2f(x^*_{n+1})}{f'(x_n) + f'(x^*_{n+1})}, \tag{29}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + 4h\frac{f(x^*_{n+1})}{f'(x_n) + f'(x^*_{n+1})} + 4h^2\frac{f(x^*_{n+1})f'(x^*_{n+1})}{[f'(x_n) + f'(x^*_{n+1})]^2} - 4h^2\ \frac{f^2(x^*_{n+1})f''(x^*_{n+1})}{[f'(x_n) + f'(x^*_{n+1})]^3}, \tag{30}$$

where

$$x^*_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Now, the following question arises. Is there a value of h such that the iteration methods defined by (29) and (30) has a maximum order of convergence?

The preceding question can be answered as follows.

Theorem 3.1 *Let $\alpha \in I$ be a simple zero of a sufficiently differentiable function $f : I \to \mathbf{R}$ for an open interval I. Then the method defined by* (29) *has a maximum order of convergence equal to three for $h = -1$, and it then satisfies the following error equation:*

$$e_{n+1} = 2c_2^3 e_n^3 + O(e_n^4). \tag{31}$$

The method defined by (30) *has a maximum order of convergence equal to four for $h = -1$, and it then satisfies the following error equation*

$$e_{n+1} = c_2^3 e_n^4 + O(e_n^5), \tag{32}$$

where $e_n = x_n - \alpha$ and $c_k = f^{(k)}(\alpha)/k!f'(\alpha)$.

Proof Let α be a simple zero of f. Consider the iteration function F defined by

$$F(x) = x - \frac{f(x)}{f'(x)} + h\,\frac{2f(z(x))}{f'(x) + f'(z(x))}, \tag{33}$$

where $z(x) = x - \dfrac{f(x)}{f'(x)}$. From the Taylor expansion of $F(x_n)$ around $x = \alpha$, we obtain

$$\begin{aligned} x_{n+1} = F(x_n) &= \alpha + (1+h)\frac{f''(\alpha)}{2f'(\alpha)}e_n^2 \\ &\quad - \left[2(2+3h)\left(\frac{f''(\alpha)}{2f'(\alpha)}\right)^3 - 2(1+h)\frac{f^{(3)}(\alpha)}{6f'(\alpha)}\right]e_n^3 + O(e_n^4), \end{aligned} \tag{34}$$

where $e_n = x_n - \alpha$. Thus,

$$e_{n+1} = (1+h)c_2 e_n^2 - 2\left[(2+3h)c_2^3 - (1+h)c_3\right]e_n^3 + O(e_n^4), \tag{35}$$

where $c_k = f^{(k)}(\alpha)/k!f'(\alpha)$. This last equation establishes the maximum order of convergence for $h = -1$; since $e_{n+1} = 2c_2^3 e_n^3 + O(e_n^4)$, the local order is three and there is no other iteration method from (29), varying the values of h, with order greater than or equal to three.

Consider the iteration function G defined by

$$\begin{aligned} G(x) = x - \frac{f(x)}{f'(x)} &+ 4h\frac{f(z(x))}{f'(x) + f'(z(x))} + 4h^2\frac{f(z(x))f'(z(x))}{[f'(x) + f'(z(x))]^2} \\ &- 4h^2\frac{f^2(z(x))f''(z(x))}{[f'(x) + f'(z(x))]^3}. \end{aligned} \tag{36}$$

By the Taylor expansion of $G(x_n)$ around $x = \alpha$, we obtain

$$\begin{aligned} x_{n+1} &= \alpha + (1+h)^2\frac{f''(\alpha)}{2f'(\alpha)}e_n^2 \\ &\quad -2(1+h)\left[(1+2h)\left(\frac{f''(\alpha)}{2f'(\alpha)}\right)^2 - (1+h)\frac{f^{(3)}(\alpha)}{6f'(\alpha)}\right]e_n^3 \\ &\quad + \Bigg[(4+14h+11h^2)\left(\frac{f''(\alpha)}{2f'(\alpha)}\right)^3 - 7(1+h)(1+2h)\frac{f''(\alpha)}{2f'(\alpha)}\frac{f^{(3)}(\alpha)}{6f'(\alpha)} \\ &\quad + 3(1+h)^2\frac{f^{(4)}(\alpha)}{24f'(\alpha)}\Bigg]e_n^4 + O(e_n^5). \end{aligned} \tag{37}$$

Thus,

$$\begin{aligned}e_{n+1} = (1+h)^2 c_2 e_n^2 - 2(1+h)[(1+2h)c_2^2 - (1+h)c_3]e_n^3 \\ +[(4+14h+11h^2)c_2^3 - 7(1+h)(1+2h)c_2c_3 + 3(1+h)^2c_4]e_n^4 \\ +O(e_n^5).\end{aligned} \tag{38}$$

This last equation establishes the maximum order of convergence for $h = -1$; since $e_{n+1} = c_2^3 e_n^4 + O(e_n^5)$, the local order is four and there is no other iteration method from (30), varying the values of h, with order greater than or equal to four. □

With different ξ, for example, $\xi = \gamma$, we have the approximations

$$x \approx \gamma + h\frac{f(\gamma)}{f'(\gamma)}, \tag{39}$$

$$x \approx \gamma + h(2+h)\frac{f(\gamma)}{f'(\gamma)} - h^2\frac{f^2(\gamma)f''(\gamma)}{2f'^3(\gamma)}, \tag{40}$$

for which we obtain the iterative methods

$$x_{n+1} = x_n + h\frac{f(x_n)}{f'(x_n)}, \tag{41}$$

$$x_{n+1} = x_n + h(2+h)\frac{f(x_n)}{f'(x_n)} - h^2\frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}. \tag{42}$$

When $h = -1$, (41) is the Newton iteration formula and (42) is the exactly the same as that given in [14].

Proceeding as in the proof of Theorem 3.1 for the iteration formulae (41) and (42), we obtain the following error equations

$$e_{n+1} = (1+h)e_n - hc_2e_n^2 + O(e_n^3) \tag{43}$$

and

$$\begin{aligned}e_{n+1} = (1+h)^2 e_n - 2h(1+h)c_2e_n^2 - h[(4+5h)c_3 - 2(2+3h)c_2^2]e_n^3 \\ -h[3(2+3h)c_4 - 2(7+13h)c_2c_3 + (8+17h)c_2^3]e_n^4 + O(e_n^5),\end{aligned} \tag{44}$$

respectively.

Therefore, the method defined by (41) has a maximum order of convergence equal to two for $h = -1$, and it then satisfies the error equation:

$$e_{n+1} = c_2e_n^2 + O(e_n^3). \tag{45}$$

The method defined by (42) has a maximum order of convergence equal to three for $h = -1$, and it then satisfies the following error equation:

$$e_{n+1} = (2c_2^2 - c_3)e_n^3 + O(e_n^4). \tag{46}$$

Thus, we have proved that the iterative method defined by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)} \tag{47}$$

has order of convergence three.

In a similar fashion as in the above, one can continue to derive a sequence of new iteration methods or improving the already existing iteration formulae by specifying appropriating values for ξ.

As our second special case of the form (2), we consider the following equation:

$$x + \frac{f(\gamma)}{f(x) - f(\gamma)}(x-\gamma) = \gamma. \tag{48}$$

The Eq. (48) can be derived by rewriting the first equation of the following system equivalent to (1):

$$f(\gamma) + f'(\gamma)(x-\gamma) + g(x) = 0, \tag{49}$$

$$g(x) = f(x) - f(\gamma) - f'(\gamma)(x-\gamma), \tag{50}$$

in the form

$$f(\gamma) + (x-\gamma)\left[f'(\gamma) + \frac{g(x)}{(x-\gamma)}\right] = 0, \tag{51}$$

giving

$$x + \frac{f(\gamma)(x-\gamma)}{f'(\gamma)(x-\gamma) + g(x)} = \gamma, \tag{52}$$

and then simplifying (52) using (50).

Applying (11) to (13) with $L(x) = x,\ N(x) = \dfrac{f(\gamma)}{f(x) - f(\gamma)}(x-\gamma)$, and $c = \gamma$, after some simplications, the components $v_0, v_1, v_2, v_3, \ldots$ are recursively determined as follows.

$$v_0 = \xi, \tag{53}$$

$$v_1 = h\,[\xi + N(\xi) - \gamma] = h\frac{f(\xi)}{f(\xi) - f(\gamma)}(\xi - \gamma), \tag{54}$$

$$\begin{aligned} v_2 &= (1+h)\,v_1 + h\,N'(v_0)v_1 = (1+h)h\,\frac{f(\xi)}{f(\xi) - f(\gamma)}(\xi-\gamma) \\ &\quad + h^2\,\frac{f(\gamma)f(\xi)}{[f(\xi) - f(\gamma)]^2}(\xi-\gamma) - h^2\,\frac{f(\gamma)f(\xi)f'(\xi)}{[f(\xi) - f(\gamma)]^3}(\xi-\gamma)^2, \end{aligned} \tag{55}$$

and so on.

Therefore, proceeding as in the first case with $\xi = \gamma - \dfrac{f(\gamma)}{f'(\gamma)}$, we arrive at the following iterative methods

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \tag{56}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + h\frac{f(x_n)f(x^*_{n+1})}{[f(x_n) - f(x^*_{n+1})]f'(x_n)}, \tag{57}$$

$$\begin{aligned} x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} + h(2+h)\frac{f(x_n)f(x^*_{n+1})}{[f(x_n) - f(x^*_{n+1})]f'(x_n)} \\ &\quad - h^2\frac{f^2(x_n)f(x^*_{n+1})}{[f(x_n) - f(x^*_{n+1})]^2 f'(x_n)} + h^2\,\frac{f^3(x_n)f(x^*_{n+1})f'(x^*_{n+1})}{[f(x_n) - f(x^*_{n+1})]^3 f'^2(x_n)}, \end{aligned} \tag{58}$$

where

$$x_{n+1}^{*} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

By the help of Maple, we have as the error equation for (57)

$$e_{n+1} = (1+h)c_2e_n^2 - \left[(2+3h)c_2^2 - 2(1+h)c_3\right]e_n^3 + O(e_n^4), \tag{59}$$

and as the error equation for (58)

$$\begin{aligned} e_{n+1} = {} & (1+h)^2c_2e_n^2 - 2(1+h)[(1+2h)c_2^2 - (1+h)c_3]e_n^3 \\ & +[(4+14h+11h^2)c_2^3 - (1+h)(7+13h)c_2c_3 + 3(1+h)^2c_4]e_n^4 \\ & +O(e_n^5), \end{aligned} \tag{60}$$

where $e_n = x_n - \alpha$ and $c_k = f^{(k)}(\alpha)/k!f'(\alpha)$.
Equations (59) and (60) establish the maximum orders of convergence for $h = -1$; since $e_{n+1} = c_2^2e_n^3 + O(e_n^4)$ and $e_{n+1} = c_2^3e_n^4 + O(e_n^5)$, the local order are three and four, respectively, and there are no other iteration method from (57) and (58), varying the values of h, with order greater than or equal to three and four, respectively.

Thus, we have the following convergence theorem.

Theorem 3.2 *Let $\alpha \in I$ be a simple zero of a sufficiently differentiable function $f : I \to \mathbf{R}$ for an open interval I. Then the method defined by* (57) *has a maximum order of convergence equal to three for $h = -1$, and it then satisfies the following error equation*

$$e_{n+1} = c_2^2e_n^3 + O(e_n^4), \tag{61}$$

where, $e_n = x_n - \alpha$ and $c_k = f^{(k)}(\alpha)/k!f'(\alpha)$. The method defined by (58) *has a maximum order of convergence equal to four for $h = -1$, and it then satisfies the following error equation*

$$e_{n+1} = c_2^3e_n^4 + O(e_n^5). \tag{62}$$

We note that the methods (57) and (58) do not require the computation of second or higher derivatives to carry out iterations. By taking different values for ξ, one can continue the above process to derive a sequence of new iteration methods or improving the existing iteration formulae.

We next show that any iteration method gives rise to an equivalent system to (1) of the form (2), so that the corresponding iteration function can be used to construct Newton-like methods. To that end let us consider the following equivalent system to (1) by the Newton iteration, for the sake of simplicity,

$$x - \phi(x) = 0, \tag{63}$$

$$\phi(x) = x - \frac{f(x)}{f'(x)}. \tag{64}$$

Applying (11) to (13) with $L(x) = x,\ N(x) = -\phi(x) = -x + \dfrac{f(x)}{f'(x)}$, and $c = 0$, after some simplications, the components $v_0, v_1, v_2, v_3, \ldots$ are recursively determined as follows.

$$v_0 = \xi, \tag{65}$$

$$v_1 = h\,\frac{f(\xi)}{f'(\xi)}, \tag{66}$$

$$v_2 = (1+h)h\,\frac{f(\xi)}{f'(\xi)} - h^2\,\frac{f^2(\xi)f''(\xi)}{f'^3(\xi)}, \tag{67}$$

and so on.
Therefore, proceeding as in the previous cases with $\xi = \gamma - \dfrac{f(\gamma)}{f'(\gamma)}$, we can obtain the following iterative methods

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{68}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + h\,\frac{f(x^*_{n+1})}{f'(x^*_{n+1})}, \tag{69}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + h(2+h)\frac{f(x^*_{n+1})}{f'(x^*_{n+1})} - h^2\,\frac{f^2(x^*_{n+1})f''(x^*_{n+1})}{f'^3(x^*_{n+1})}, \tag{70}$$

where $x^*_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$.

It should be pointed out that setting $h = -1$ in the iteration formula (69) reduces to the well-known double-Newton method, that is, the composite iteration method of two Newton methods, that converges with the fourth-order [32].
By the help of Maple we have the following error equations

$$\begin{aligned} e_{n+1} &= (1+h)c_2e_n^2 + 2(1+h)(c_3 - c_2^2)e_n^3 \\ &\quad + [3(1+h)c_4 - 7(1+h)c_2c_3 + (4+3h)c_2^3]e_n^4 + O(e_n^5), \end{aligned} \tag{71}$$

and

$$\begin{aligned} e_{n+1} &= (1+h)^2c_2e_n^2 + 2(1+h)^2(c_3 - c_2^2)e_n^3 \\ &\quad + [3(1+h)^2c_4 - 7(1+h)^2c_2c_3 + (4+6h+h^2)c_2^3]e_n^4 + O(e_n^5), \end{aligned} \tag{72}$$

for (69) and (70), respectively.

Thus, we have the following convergence result.

Theorem 3.3 *Let $\alpha \in I$ be a simple zero of a sufficiently differentiable function $f : I \to \mathbf{R}$ for an open interval I. Then the method defined by* (69) *has a maximum order of convergence equal to four for $h = -1$, and it then satisfies the error equation:*

$$e_{n+1} = c_2^3e_n^4 + O(e_n^5). \tag{73}$$

The method defined by (70) *has a maximum order of convergence equal to four for* $h = -1$, *and it then satisfies the following error equation*

$$e_{n+1} = -c_2^3 e_n^4 + O(e_n^5), \tag{74}$$

where $e_n = x_n - \alpha$ *and* $c_k = f^{(k)}(\alpha)/k! f'(\alpha)$.

It should be noted at this point that using the other iteration functions instead of the Newton iteration function in (64) will also result in many of other Newton-like iteration formulae. Again, one can continue the above process to derive a sequence of new iteration methods or improving the already existing iteration formulae by taking different values for ξ.

As our fourth special case of the form (2) let us consider the following coupled system equivalent to (1):

$$x + \frac{g(x)}{f'(\gamma)} = \gamma - \frac{f(\gamma)}{f'(\gamma)}, \tag{75}$$

$$g(x) = f(x) - f(\gamma) - f'(\gamma)(x - \gamma). \tag{76}$$

Applying (11) to (14) with $L(x) = x$, $N(x) = \dfrac{g(x)}{f'(\gamma)}$, and $c = \gamma - \dfrac{f(\gamma)}{f'(\gamma)}$, the components $v_0, v_1, v_2, v_3, \ldots$ are recursively determined as follows.

$$v_0 = \xi, \tag{77}$$

$$v_1 = h\,[\xi + N(\xi) - c] = h\left[\xi + \frac{g(\xi)}{f'(\gamma)} - \gamma + \frac{f(\gamma)}{f'(\gamma)}\right] = h\,\frac{f(\xi)}{f'(\gamma)}, \tag{78}$$

$$v_2 = (1+h)v_1 + h\,N'(v_0)v_1 = h\,\frac{f(\xi)}{f'(\gamma)} + h^2\frac{f(\xi)f'(\xi)}{f'^2(\gamma)}, \tag{79}$$

$$\begin{aligned} v_3 &= (1+h)v_2 + h\left[N'(v_0)v_2 + \frac{1}{2!}N''(v_0)v_1^2\right] \\ &= h\frac{f(\xi)}{f'(\gamma)} + 2h^2\frac{f(\xi)f'(\xi)}{f'^2(\gamma)} + h^3\frac{f(\xi)[f(\xi)f''(\xi) + 2f'^2(\xi)]}{2f'^3(\gamma)}, \end{aligned} \tag{80}$$

and so on. This case has been investigated in [11] when $h = -1$, by using the homotopy perturbation method. Here, we give more general iteration formula using the recursive relation (11) to (15) and a related convergence analysis.
With $\xi = c = \gamma - \dfrac{f(\gamma)}{f'(\gamma)}$, we can obtain the iterative methods

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \tag{81}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + h\,\frac{f(x^*_{n+1})}{f'(x_n)}, \tag{82}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + 2h\,\frac{f(x^*_{n+1})}{f'(x_n)} + h^2\frac{f(x^*_{n+1})f'(x^*_{n+1})}{f'^2(x_n)}, \tag{83}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + 3h\,\frac{f(x^*_{n+1})}{f'(x_n)} + 3h^2\,\frac{f(x^*_{n+1})f'(x^*_{n+1})}{f'^2(x_n)} + h^3\,\frac{f(x^*_{n+1})\left[f(x^*_{n+1})f''(x^*_{n+1}) + 2f'^2(x^*_{n+1})\right]}{2f'^3(x_n)}, \quad (84)$$

where

$$x^*_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

It should be mentioned that setting $h = -1$ in the iteration formula (82) reduces to a modification of Newton's method due to Potra and Pták, the two-step method [6] that converges cubically in some neighborhood of α. Setting $h = -1$ in the iteration formula (83) reduces to the method derived in [10] by applying the Adomian decomposition method.
Again, by the help of Maple, we can obtain the following error equations

$$e_{n+1} = (1+h)c_2e_n^2 - 2[(1+2h)c_2^2 - (1+h)c_3]e_n^3 + O(e_n^4), \quad (85)$$

and

$$e_{n+1} = (1+h)^2c_2e_n^2 - 2(1+h)[(1+3h)c_2^2 - (1+h)c_3]e_n^3 + [(4+26h+27h^2)c_2^3 - 7(1+h)(1+3h)c_2c_3 + 3(1+h)^2c_4]e_n^4 + O(e_n^5), \quad (86)$$

for (82) and (83), respectively.

Thus, we have the following result.

Theorem 3.4 *Let $\alpha \in I$ be a simple zero of a sufficiently differentiable function $f : I \to \mathbf{R}$ for an open interval I. Then the method defined by* (82) *has a maximum order of convergence equal to three for $h = -1$, and it then satisfies the following error equation*

$$e_{n+1} = 2c_2^2e_n^3 + O(e_n^4), \quad (87)$$

where, $e_n = x_n - \alpha$ and $c_k = f^{(k)}(\alpha)/k!f'(\alpha)$. The method defined by (83) *has a maximum order of convergence equal to four for $h = -1$, and it then satisfies the following error equation*

$$e_{n+1} = 5c_2^3e_n^4 + O(e_n^5). \quad (88)$$

When $\xi = \gamma$, it can be shown that the third-order approximation yields the iterative method defined by

$$x_{n+1} = x_n + h(3+3h+h^2)\frac{f(x_n)}{f'(x_n)} + h^3\frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}, \quad (89)$$

and that by the help of Maple, the method (89) has a maximum order of convergence equal to three for $h = -1$, it satisfies the following error equation:

$$e_{n+1} = (2c_2^2 - c_3)e_n^3 + O(e_n^4). \quad (90)$$

We remark that the iteration formula (89) with $h = -1$ reduces to the Householder's iteration [18] known to be with cubic convergence.

In a similar fashion as in the above, one can continue to derive a sequence of new iteration methods or improving the already existing iteration methods by specifying appropriate values for γ and ξ, whether these values are freely selected or not.

4 Examples

The order of convergence is defined by p such that

$$\lim_{n\to\infty}\frac{|e_{n+1}|}{|e_n|^p}=c\neq 0.$$

Then the computational order of convergence (COC) ρ can be approximated using the formula

$$\rho\approx\frac{\ln|(x_{n+1}-\alpha)/(x_n-\alpha)|}{\ln|(x_n-\alpha)/(x_{n-1}-\alpha)|}.$$

All computations were done using MAPLE using 64 digit floating point arithmetics (Digits:=64). We accept an approximate solution rather than the exact root, depending on the precision (ϵ) of the computer. We use the following stopping criteria for computer programs: $(i)\ |x_{n+1}-x_n|<\epsilon,\ (ii)\ |f(x_{n+1})|<\epsilon$, and so, when the stopping criterion is satisfied, x_{n+1} is taken as the exact root α computed. For numerical illustrations in this section we used the fixed stopping criterion $\epsilon=10^{-15}$.

We present some numerical test results for various cubically convergent iterative schemes in Table 1. Compared were the Newton method(NM), the method of Weerakoon and Fernando [33](WF) defined by

$$x_{n+1}=x_n-\frac{2f(x_n)}{f'(x_n)+f'(x_n-f(x_n)/f'(x_n))},$$

the method derived from midpoint rule [12] (MP) defined by

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n-f(x_n)/(2f'(x_n)))},$$

the method of Abbasbandy [1](AM) defined by

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}-\frac{f^2(x_n)f''(x_n)}{2f'^3(x_n)}-\frac{f^3(x_n)f''^2(x_n)}{2f'^5(x_n)},$$

the method of Homeier [17](HM) defined by

$$x_{n+1}=x_n-\frac{f(x_n)}{2}\left(\frac{1}{f'(x_n)}+\frac{1}{f'(x_n-f(x_n)/f'(x_n))}\right),$$

and the methods (29) with $h=-1$ (CM1) and (57) with $h=-1$ (CM2) introduced in the present contribution. We used the same test functions as Weerakoon and Fernando [33]

$$f_1(x) = x^3 + 4x^2 - 10,$$
$$f_2(x) = \sin^2 x - x^2 + 1,$$
$$f_3(x) = x^2 - e^x - 3x + 2,$$
$$f_4(x) = \cos x - x,$$
$$f_5(x) = (x-1)^3 - 1,$$
$$f_6(x) = x^3 - 10,$$
$$f_7(x) = xe^{x^2} - \sin^2 x + 3\cos x + 5,$$
$$f_8(x) = e^{x^2+7x-30} - 1.$$

As convergence criterion, it was required that the distance of two consecutive approximations δ for the zero was less than 10^{-15}. Also displayed are the number of iterations to approximate the zero (IT), the number of functional evaluations (NFE) counted as the sum of the number of evaluations of the function itself plus the number of evaluations of the derivative, the computational order of convergence (COC), the approximate zero x_*, and the value $f(x_*)$. Note that the approximate zeroes were displayed only up to the 28th decimal places, so it making all looking the same though they may in fact differ.

We also present some numerical test results for various quartically convergent iterative schemes in Table 2. Compared were the double-Newton method [32](DN) defined by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f(x_n - f(x_n)/f'(x_n))}{f'(x_n - f(x_n)/f'(x_n))},$$

the Ostrowski method [30] (OM) defined by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{f(x_n)f(x_n - f(x_n)/f'(x_n))}{f'(x_n)[2f(x_n - f(x_n)/f'(x_n)) - f(x_n)]},$$

an iteration method known to converge with the fourth-order [32](TM) defined by

$$x_{n+1} = x_n - \frac{1}{4}\left[\frac{f(x_n)}{f'(x_n)} + \frac{3f(x_n)}{f'\{x_n - (2/3)f(x_n)/f'(x_n - (1/3)f(x_n)/f'(x_n))\}}\right],$$

and the methods (58) with $h = -1$ (CM3), (70) with $h = -1$ (CM4) and (83) with $h = -1$ (CM5) obtained in the present contribution.

The test results in Tables 1 and 2 show that the computational orders of convergence of the proposed methods (CM1–CM5) are in accordance with the theory developed in the previous section. For most of the functions we tested, the methods introduced in the present presentation for numerical tests behave equal or better performance compared to the other methods of order three or four. The important characteristic of the methods CM1–CM3 and CM5 is that they do not require the computation of second-order or higher-order derivatives of the function to carry out iterations.

Table 1 Comparison of various cubically convergent iterative schemes and the Newton method

	IT	NFE	COC	x_*	$f(x_*)$	δ
$f_1, x_0 = -0.3$						
NM	55	110	2	1.3652300134140968457608068290	1.95e–60	4.92e–31
WF	7	21	3	1.3652300134140968457608068290	0.0e–01	6.25e–26
MP	19	57	3	1.3652300134140968457608068290	0.0e–01	4.03e–43
AM	22	66	2.97	1.3652300134140968457608068290	–1.47e–57	1.14e–19
HM	87	261	3	1.3652300134140968457608068290	0.0e–01	3.06e–46
CM1	13	52	3	1.3652300134140968457608068290	0.0e–01	4.83e–28
CM2	5	15	3	1.3652300134140968457608068290	7.15e–53	2.62e–18
$f_2, x_0 = 1$						
NM	7	14	2	1.4044916482153412260350868178	–1.04e–50	7.33e–26
WF	5	15	3	1.4044916482153412260350868178	–2.0e–63	3.79e–30
MP	5	15	3	1.4044916482153412260350868178	1.3e–63	7.7e–33
AM	5	15	2.86	1.4044916482153412260350868178	–5.81e–55	1.39e–18
HM	4	12	3.01	1.4044916482153412260350868178	–5.4e–62	7.92e–21
CM1	5	20	3	1.4044916482153412260350868178	–2.0e–63	2.49e–33
CM2	5	15	3	1.4044916482153412260350868178	–2.0e–63	7.14e–32
$f_3, x_0 = 2$						
NM	6	12	2	0.25753028543986076045536730494	2.93e–55	9.1e–28
WF	5	15	3	0.25753028543986076045536730494	1.0e–63	1.62e–34
MP	4	12	3.01	0.25753028543986076045536730494	1.0e–63	3.95e–24
AM	5	15	3	0.25753028543986076045536730494	1.0e–63	1.45e–26
HM	5	15	3	0.25753028543986076045536730494	0.0e–01	9.33e–43
CM1	4	16	2.96	0.25753028543986076045536730494	–6.4e–50	1.25e–16
CM2	4	12	2.98	0.25753028543986076045536730494	–3.18e–53	9.88e–18
$f_4, x_0 = 1.7$						
NM	5	10	2	0.73908513321516064165531208767	–2.03e–32	2.34e–16
WF	4	12	3.01	0.73908513321516064165531208767	1.0e–64	1.04e–21
MP	4	12	2.99	0.73908513321516064165531208767	–3.32e–61	1.45e–20
AM	4	12	3.01	0.73908513321516064165531208767	–7.14e–47	8.6e–16
HM	4	12	3	0.73908513321516064165531208767	–5.02e–59	9.64e–20
CM1	4	16	3	0.73908513321516064165531208767	1.0e–64	6.68e–28
CM2	4	12	3	0.73908513321516064165531208767	0.0e–01	1.07e–26
$f_5, x_0 = 3.5$						
NM	8	16	2	2	2.06e–42	8.28e–22
WF	6	18	3	2	0.0e–01	3.28e–37
MP	6	18	3	2	0.0e–01	1.26e–42
AM	5	15	2.99	2	0.0e–01	4.3e–22
HM	5	15	3.0	2	0.0e–01	1.46e–24
CM1	6	24	3	2	0.0e–01	5.41e–40
CM2	6	18	3	2	0.0e–01	1.10e–40
$f_6, x_0 = 1.5$						
NM	7	14	2	2.1544346900318837217592935665	2.06e–54	5.64e–28
WF	5	15	3	2.1544346900318837217592935665	–5.0e–63	5.08e–32
MP	5	15	3	2.1544346900318837217592935665	–5.0e–63	7.04e–37
AM	5	15	2.99	2.1544346900318837217592935665	–5.0e–63	1.18e–25
HM	4	12	3	2.1544346900318837217592935665	–5.0e–63	9.8e–23
CM1	5	20	3	2.1544346900318837217592935665	–5.0e–63	4.09e–37
CM2	5	15	3	2.1544346900318837217592935665	–5.0e–63	4.57e–35
$f_7, x_0 = -2$						
NM	9	18	2	–1.2076478271309189270094167584	–2.27e–40	2.73e–21
WF	7	21	3	–1.2076478271309189270094167584	–4.0e–63	3.11e–44
MP	6	18	3	–1.2076478271309189270094167584	–4.0e–63	2.12e–23
AM	6	18	3	–1.2076478271309189270094167584	–4.0e–63	4.35e–45
HM	6	18	3	–1.2076478271309189270094167584	–4.0e–63	2.57e–32
CM1	6	24	3	–1.2076478271309189270094167584	–2.86e–53	8.55e–19
CM2	6	18	3	–1.2076478271309189270094167584	–1.98e–57	3.51e–20

Table 1 continued

	IT	NFE	COC	x_*	$f(x_*)$	δ
$f_8, x_0 = 3.5$						
NM	13	26	2	3	1.52e–47	4.2e–25
WF	9	27	3	3	0.0e–01	4.07e–25
MP	8	24	2.99	3	6.57e–48	2.41e–17
AM	7	21	2.98	3	–4.33e–48	2.25e–17
HM	8	24	3	3	2.0e–62	2.43e–33
CM1	9	36	3	3	0.0e–01	3.93e–34
CM2	9	27	3	3	0.0e–01	5.75e–39

In Theorems 3.1 to 3.4 we proved that the iteration formulas presented in this work have the maximum order of convergence when $h = -1$. However, it should be emphasized that the order of convergence is a property of iteration formula near root: the order of convergence is one thing, the total number of iterations is the other. In general, for a given iteration formula, the total number of iterations depends not only on the order of convergence but also the initial approximation x_0. For iteration formulas obtained in this work, the total number of iterations also depends upon h. To illustrate the heavy dependence of the total number of iterations on h, we list the total iteration number in the seventh equation $xe^{x^2} - \sin^2 x + 3\cos x + 5 = 0$ for different initial approximation x_0 and h in Tables 3, 4 and 5.

The test results from Tables 3 to 5 show that in almost all of the cases the iteration formula with $h = -1$ gives the maximum iteration number, although it corresponds to the highest order of convergence. These illustrations indicate that, sometimes, much less iteration number is needed if a proper value of h is chosen when the initial approximation x_0 is far from root. Note that, for a given nonlinear algebraic equation, it is rather hard to choose an initial approximation near a root. It seems that there exists the optimal value of h, which corresponds the minimum number of iteration. As pointed by Liao in his book [27] it is the auxiliary parameter h that provides us with a simple way to control the convergence region and rate. The optimal choice of the parameter h will be an interesting work in future.

5 Conclusion

In this work we presented a numerical scheme, which can be used in constructing various kinds of Newton-like iteration methods improving Newton's formula or the existing iteration formulae from several transformed equations equivalent to the nonlinear equation. This turns out to be very fruitful in developing or improving Newton-like methods. These transformed equations are also constructible in many different ways equivalently to the nonlinear equations using such functions as the known iterative functions, and so this merit will make the proposed numerical scheme versatile in applications.

We have also discussed that the convergence orders and error equations of the methods can be found exactly and explicitly with the help of Maple package. The proposed scheme can be continuously applied to generate iterative methods with arbitrarily specified order of convergence or extended to systems of nonlinear equations in combination of symbolic computation of mathematical software

Table 2 Comparison of various quartically convergent iterative schemes

	IT	NFE	COC	x_*	$f(x_*)$	δ
$f_1, x_0 = -0.3$						
DN	28	112	3.99	1.3652300134140968457608068290	0.0e–01	4.92e–31
OM	46	138	4	1.3652300134140968457608068290	0.0e–01	9.16e–57
TM	86	344	4	1.3652300134140968457608068290	0.0e–01	6.82e–45
CM3	8	32	4	1.3652300134140968457608068290	0.0e–01	8.95e–55
CM4	11	55	4	1.3652300134140968457608068290	0.0e–01	5.44e–58
CM5	27	108	4	1.3652300134140968457608068290	0.0e–01	1.82e–50
$f_2, x_0 = 1$						
DN	4	16	3.991	1.4044916482153412260350868178	–2.0e–63	7.33e–26
OM	4	12	3.995	1.4044916482153412260350868178	1.3e–63	5.64e–28
TM	4	16	3.994	1.4044916482153412260350868178	1.3e–63	3.08e–27
CM3	4	16	3.999	1.4044916482153412260350868178	–2.0e–63	2.73e–36
CM4	4	20	4.003	1.4044916482153412260350868178	1.30e–63	7.39e–31
CM5	5	20	3.951	1.4044916482153412260350868178	–2.0e–63	1.31e–17
$f_3, x_0 = 2$						
DN	4	16	4	0.25753028543986076045536730494	0.0e–01	7.74e–56
OM	4	12	3.994	0.25753028543986076045536730494	1.0e–63	2.70e–23
TM	4	16	3.998	0.25753028543986076045536730494	–1.0e–63	3.17e–34
CM3	4	16	4.001	0.25753028543986076045536730494	–1.0e–63	7.25e–44
CM4	4	20	4.004	0.25753028543986076045536730494	–1.0e–63	1.73e–39
CM5	4	16	3.978	0.25753028543986076045536730494	1.0e–63	9.46e–29
$f_4, x_0 = 1.7$						
DN	3	12	3.595	0.73908513321516064165531208767	0.0e–01	2.34e–16
OM	4	12	4	0.73908513321516064165531208767	0.0e–01	3.18e–48
TM	4	16	4	0.73908513321516064165531208767	0.0e–01	1.28e–48
CM3	4	16	4	0.73908513321516064165531208767	0.0e–01	7.28e–61
CM4	3	15	3.589	0.73908513321516064165531208767	0.0e–01	1.81e–16
CM5	4	16	4	0.73908513321516064165531208767	0.0e–01	1.87e–53
$f_5, x_0 = 3.5$						
DN	5	20	4	2	0.0e–01	6.86e–43
OM	5	15	4	2	0.0e–01	2.21e–49
TM	5	20	4	2	0.0e–01	1.08e–46
CM3	5	20	4	2	0.0e–01	3.82e–41
CM4	4	20	3.997	2	0.0e–01	3.47e–31
CM5	5	20	3.989	2	0.0e–01	2.74e–24
$f_6, x_0 = 1.5$						
DN	4	16	3.996	2.1544346900318837217592935665	–5.0e–63	5.64e–28
OM	4	12	3.998	2.1544346900318837217592935665	–5.0e–63	3.73e–32
TM	4	16	3.997	2.1544346900318837217592935665	–5.0e–63	1.44e–30
CM3	4	16	3.999	2.1544346900318837217592935665	–5.0e–63	1.63e–33
CM4	4	20	4.002	2.1544346900318837217592935665	1.0e–62	2.08e–31
CM5	5	20	3.985	2.1544346900318837217592935665	–5.0e–63	1.57e–22
$f_7, x_0 = -2$						
DN	5	20	3.997	–1.2076478271309189270094167584	–4.0e–63	2.73e–21
OM	5	15	4	–1.2076478271309189270094167584	–4.0e–63	3.71e–43
TM	5	20	4.001	–1.2076478271309189270094167584	–4.0e–63	7.32e–28
CM3	7	28	3	–1.2076478271309189270094167584	–4.0e–63	2.60e–26
CM4	5	25	4	–1.2076478271309189270094167584	–4.0e–63	1.49e–45
CM5	7	28	2	–1.2076478271309189270094167584	1.67e–35	5.24e–19

package such as Maple. Also the way that the numerical scheme is constructed may be replaced by any other perturbation method such as the Homotopy perturbation method which, if successful, may result in another numerical scheme for constructing the Newton-like methods.

Table 3 Total iteration number of (29) for different x_0 and h

x_0	$h = -1$	$h = -2$	$h = -3$	$h = -4$
–5	21	18	16	17
–10	70	55	46	42
–15	152	114	94	83
–20	266	200	161	136
5	588	35	26	21
20	271	201	186	134

Table 4 Total iteration number of (57) for different x_0 and h

x_0	$h = -1$	$h = -2$	$h = -3$	$h = -4$
–5	20	17	16	14
–10	68	53	44	37
–15	148	111	89	78
–20	259	192	153	9574
5	44	21	150	4429
20	Divergence	195	304	436

Table 5 Total iteration number of (69) for different x_0 and h

x_0	$h = -1$	$h = -2$	$h = -3$	$h = -4$
–5	16	14	15	Divergence
–10	54	40	33	Divergence
–15	117	81	64	52
–20	205	141	108	88
5	2119	873	224	Divergence
20	793	145	2875	547

Acknowledgements The author would like to thank the referees for their many useful comments and constructive suggestions which substantially improved the manuscript.

References

1. Abbasbandy, S.: Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method. Appl. Math. Comput. **145**, 887–893 (2003)
2. Abbasbandy, S.: Modified homotopy perturbation method for nonlinear equations and comparison with Adomian decomposition method. Appl. Math. Comput. **172**(1), 431–438 (2006)
3. Adomian, G.: Nonlinear stochastic differential equations. J. Math. Anal. Appl. **55**(3), 441–452 (1976)
4. Adomian, G., Adomian, G.E.: A global method for solution of complex systems. Math. Model. Anal. **5**(3), 521–568 (1984)
5. Adomain, G.: Solving Frontier problems of Physics: The Decomposition Method. Kluwer, Boston and London, (1994)
6. Amat, S., Busquier, S., Gutiérrez, J.M.: Geometric constructions of iterative functions to solve nonlinear equations. J. Comput. Appl. Math. **157**, 197–205 (2003)
7. Argyros, I.K.: Results on Newton methods: Part I. A unified approach for constructing perturbed Newton-like methods in Banach space and their applications. Appl. Math. Comput. **74**, 119–141 (1996)
8. Babolian, E., Biazar, J.: Solution of nonlinear equations by modified adomian decomposition method. Appl. Math. Comput. **132**, 167–172 (2002)
9. Basto, M., Semiao, V., Calheiros, F.L.: A new iterative method to compute nonlinear equations. Appl. Math. Comput. **173**(1), 468–483 (2006)

10. Chun, C.: Iterative methods improving Newton's method by the decomposition method. Comput. Math. Appl. **50**, 1559–1568 (2005)
11. Chun, C., Ham, Y.: Newton-like iteration methods for solving nonlinear equations. Commun. Numer. Methods Eng. **22**, 475–487 (2006)
12. Frontini, M., Sormani, E.: Some variant of Newton's method with third-order convergence. Appl. Math. Comput. **140**, 419–426 (2003)
13. Grau, M., Noguera, M.: A variant of Cauchy's method with accelerated fifth-order convergence. Appl. Math. Lett. **17**, 87–93 (2000)
14. He, J.H.: Newton-like iteration method for solving algebraic equations. Commun. Nonlinear Sci. Numer. Simul. **3**(2), 106–109 (1998)
15. He, J.H.: Homotopy perturbation technique for nonlinear problems. Comput. Methods Appl. Mech. Eng. **178**, 257–262 (1999)
16. He, J.H.: A coupling method of homotopy technique and perturbation technique for nonlinear problems. Int. J. Non-linear Mech. **35**(1), 37–43 (2000)
17. Homeier, H.H.H.: On Newton-type methods with cubic convergence. J. Comput. Appl. Math. **176**, 425–432 (2005)
18. Householder, A.S.: The Numerical Treatment of a Single Nonlinear Equation. McGraw-Hill, New York (1970)
19. Huang, Z.: On the error estimates of several Newton-like methods. Appl. Math. Comput. **106**, 1–16 (1999)
20. Liao, S.J.: The proposed homotopy analysis technique for the solution of nonlinear problems. Ph.D. Thesis, Shanghai Jiao Tong university (1992)
21. Liao, S.J.: An approximate solution technique not depending on small parameters: a special example. Int. J. Non-linear Mech. **30**(2), 371–380 (1995)
22. Liao, S.J.: A kind of approximate solution technique which does not depend upon small parameters (II): an application in fluid mechanics. Int. J. Non-linear Mech. **32**, 815–822 (1997)
23. Liao, S.J.: Boundary element method for general nonlinear differential operators. Eng. Anal. Bound. Elem. **20**(2), 91–99 (1997)
24. Liao, S.J.: Homotopy analysis method: an analytic technique not depending on small parameters. Shanghai J. Mech. **18**(3), 196–200 (1997)
25. Liao, S.J.: A simple way to enlarge the convergence region of perturbation approximations. Int. J. Non-linear Dyn. **19**(2), 93–110 (1999)
26. Liao, S.J.: An explicit, totally analytic approximation of Blasius viscous flow problems. Int. J. Non-linear Mech. **34**(4), 759–778 (1999)
27. Liao, S.J.: Beyond Perturbation: Introduction to Homotopy Analysis Method. Chapman & Hall/ CRC Press, Boca Raton (2003)
28. Liao, S.J.: On the homotopy analysis method for nonlinear problems. Appl. Math. Comput. **147**, 499–513 (2004)
29. Liao, S.J.: Comparison between the homotopy analysis method and homotopy perturbation method. Appl. Math. Comput. **169**(2), 1186–1194 (2005)
30. Ostrowski, A.M.: Solutions of equations and system of equations. Academic, New York, (1960)
31. Petkovic, L., et al.: On the construction of simultaneous methods for multiple zeros. Nonlinear Anal. **30**, 669–676 (1997)
32. Traub, J.F.: Iterative methods for the solution of equations, Chelsea Publishing company. New York (1982)
33. Weerakoon, S., Fernando, G.I.: A variant of Newton's method with accelerated third-order convergence. Appl. Math. Lett. **17**, 87–93 (2000)