Comparative study of eighth-order methods for finding simple roots of nonlinear equations

Changbum Chun & Beny Neta

Numerical Algorithms

ISSN 1017-1398
Volume 74
Number 4

Numer Algor (2017) 74:1169-1201
DOI 10.1007/s11075-016-0191-y

Springer

Numer Algor (2017) 74:1169–1201
DOI 10.1007/s11075-016-0191-y

ORIGINAL PAPER

Comparative study of eighth-order methods for finding simple roots of nonlinear equations

Changbum Chun[1] · Beny Neta[2]

Received: 6 May 2016 / Accepted: 5 August 2016 / Published online: 17 August 2016

Abstract Recently, there were many papers discussing the basins of attraction of various methods and ideas how to choose the parameters appearing in families of methods and weight functions used. Here, we collected many of the eighth-order schemes scattered in the literature and presented a quantitative comparison. We have used the average number of function evaluations per point, the CPU time, and the number of black points to compare the methods. Based on seven examples, we found that the best method based on the three criteria is SA8 due to Sharma and Arora.

Keywords Iterative methods · Nonlinear equations · Simple roots · Order of convergence · Extraneous fixed points · Basin of attraction

1 Introduction

There are many iterative methods for the solution of a single nonlinear equation [1]. Most are for simple roots and a few are for a repeated root. Here, we are only interested in methods for simple roots. In fact, we will not discuss derivative-free methods or methods with memory. There are many new methods and families of methods, some of which are just rediscovery of old ones or special cases of known families of methods, see, e.g., [2] for examples of such cases.

✉ Beny Neta
bneta@nps.edu

Changbum Chun
cbchun@skku.edu

1 Department of Mathematics, Sungkyunkwan University, Suwon 16419, Republic of Korea

2 Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, USA

The usual technique of comparing a new method to existing ones is by comparing the performance on selected problems using one or two initial points or by comparing the efficiency index (see [1]). In recent work, one can find a visual comparison, by plotting the basins of attraction for the methods. The idea of using basins of attraction appeared first in Stewart [3] and followed by the works of Amat et al. [4–6] and [7], Scott et al. [8], Chun et al. [9], Chicharro et al. [10], Cordero et al. [11], Neta et al. [12, 13], Chun et al. [14–16], Argyros and Magreñan [17], and Magreñan [18]. In later works ([15, 16, 19–21]), we have introduced a more quantitative comparison, by listing the average number of iterations per point, the CPU time, and the number of points requiring 40 iterations. We have also discussed how to choose the parameters appearing in the method and/or the weight function (see, e.g., [22]).

First, we list the eighth-order methods we consider here. The first one is non-optimal, since it uses five function-evaluation per step instead of four. All the other methods are optimal in the sense of Kung and Traub [23].

(i) Neta-Johnson's non-optimal method
(ii) Neta-Petković's optimal method
(iii) Neta's optimal method
(iv) Hermite-based King optimal method
(v) Hermite-based Kung-Traub optimal method
(vi) Kung-Traub's optimal method
(vii) Hermite-based Wang-Liu optimal method
(viii) Chun-Neta optimal family of methods
(ix) Weighted Maheshwari's optimal method
(x) Thukral-Petković optimal family of methods
(xi) Bi-Ren-Wu's optimal family of methods
(xii) Lotfi et al.'s optimal family of methods
(xiii) Babajee et al.'s optimal method
(xiv) Cordero et al.'s optimal method
(xv) Wang and Liu's optimal method
(xvi) Sharma and Sharma's optimal family of methods
(xvii) Cordero et al.'s optimal family of methods
(xviii) Chun and Lee's optimal family of methods
(xix) Bi-Wu-Ren's optimal family of methods
(xx) Khan et al.'s optimal family of methods
(xxi) Kou et al.'s optimal family of methods
(xxii) Džunić and Petković optimal method
(xxiii) Sharma and Arora's optimal method
(xxiv) Cordero-Torregrosa-Vassileva's optimal method
(xxv) Džunić-Petković-Petković's optimal method
(xxvi) Geum and Kim's optimal method
(xxvii) Liu and Wang's optimal method
(xxviii) Sharma and Arora Weighted Newton optimal method
(xxix) Sharma-Guha-Gupta's optimal method
(xxx) Thukral's optimal method

We now detail all the above methods.

(i) An eighth-order (NJ8) non-optimal method based on Jarratt's method [24] due to Neta and Johnson [25] is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= x_n - \frac{f_n}{\frac{1}{6}f_n' + \frac{1}{6}f'(y_n) + \frac{2}{3}f'(\eta_n)}, \\ \eta_n &= x_n - \frac{1}{8}u_n - \frac{3}{8}\frac{f_n}{f(y_n)}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f_n'}\frac{f_n' + f'(y_n) + a_2 f'(\eta_n)}{(-1-a_2)f_n' + (3+a_2)f'(y_n) + a_2 f'(\eta_n)}, \end{aligned} \tag{1}$$

where

$$u_n = \frac{f_n}{f_n'}, \tag{2}$$

and $f_n = f(x_n)$ and similarly for the derivative.

In our experiments, we have used $a_2 = -1$. This method is not optimal since it requires two function- and three derivative-evaluation per cycle.

(ii) Neta and Petković (NP8) [26] have developed an eighth-order (NP8) optimal method based on Kung and Traub's optimal fourth-order method [23] and inverse interpolation

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'}\frac{1}{[1 - f(y_n)/f_n]^2}, \\ x_{n+1} &= x_n - \frac{f_n}{f_n'} + c_n f_n^2 - d_n f_n^3, \end{aligned} \tag{3}$$

where

$$\begin{aligned} d_n &= \frac{1}{[f(y_n) - f_n][f(y_n) - f(z_n)]}\left[\frac{y_n - x_n}{f(y_n) - f_n} - \frac{1}{f_n'}\right] \\ &\quad - \frac{1}{[f(y_n) - f(z_n)][f(z_n) - f_n]}\left[\frac{z_n - x_n}{f(z_n) - f_n} - \frac{1}{f_n'}\right], \\ c_n &= \frac{1}{f(y_n) - f_n}\left[\frac{y_n - x_n}{f(y_n) - f_n} - \frac{1}{f_n'}\right] - d_n\,[f(y_n) - f_n]. \end{aligned} \tag{4}$$

(iii) An eighth-order (N8) optimal method proposed by Neta [27] is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'}\frac{f_n + \beta f(y_n)}{f_n + (\beta - 2) f(y_n)}, \\ x_{n+1} &= x_n - u_n + \gamma f_n^2 - \rho f_n^3, \end{aligned} \tag{5}$$

where

$$\rho = \frac{\phi_y - \phi_z}{F_y - F_z},\ \gamma = \phi_y - \rho F_y,\ F_y = f(y_n) - f_n,\ F_z = f(z_n) - f_n,$$
$$\phi_y = \frac{y_n - x_n}{F_y^2} - \frac{1}{F_y f_n'},\ \phi_z = \frac{z_n - x_n}{F_z^2} - \frac{1}{F_z f_n'}. \tag{6}$$

(iv) A Hermite interpolating polynomial with King's fourth-order method (HK8).

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'} \frac{f_n + \beta f(y_n)}{f_n + (\beta - 2) f(y_n)}, \\ x_{n+1} &= z_n - \frac{H_3(z_n)}{f'(z_n)}, \end{aligned} \tag{7}$$

where $H_3(z_n)$ is given by

$$H_3(z_n) = f_n + f_n' \frac{(z_n - y_n)^2 (z_n - x_n)}{(y_n - x_n)(x_n + 2y_n - 3z_n)} + f'(z_n) \frac{(z_n - y_n)(x_n - z_n)}{x_n + 2y_n - 3z_n} - f[x_n, y_n] \frac{(z_n - x_n)^3}{(y_n - x_n)(x_n + 2y_n - 3z_n)}. \tag{8}$$

(v) A Hermite interpolation-based eighth-order (HKT8) optimal method based on Kung-Traub fourth-order method (11), see [2], is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'} \frac{1}{[1 - f(y_n)/f_n]^2}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{H_3'(z_n)}, \end{aligned} \tag{9}$$

where $H_3'(z_n)$ is given by

$$H_3'(z_n) = 2(f - f[x_n, y_n]) + f[y_n, z_n] + \frac{y_n - z_n}{y_n - x_n} (f[x_n, y_n] - f_n'). \tag{10}$$

(vi) Kung-Traub's eighth-order (KT8) method [23] based on inverse interpolation [26]. It is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f_n}{f_n'} \frac{f(y_n) f_n}{[f_n - f(y_n)]^2}, \\ x_{n+1} &= z_n - \frac{f_n}{f_n'} \frac{f_n f(y_n) f(z_n)}{[f_n - f(y_n)]^2} \frac{f_n^2 + f(y_n)[f(y_n) - f(z_n)]}{[f_n - f(z_n)]^2 [f(y_n) - f(z_n)]}. \end{aligned} \tag{11}$$

(vii) Hermite interpolation-based eighth-order (WL8) optimal method proposed by Wang and Liu [28] is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f'_n}\frac{f_n}{f_n - 2f(y_n)}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{H'_3(z_n)}, \end{aligned} \tag{12}$$

where $H'_3(z_n)$ is defined by (10). Note that the first two sub-steps are Ostrowski's method [29].

(viii) Chun-Neta family (CN8) of optimal methods of eighth order [20]

$$\begin{cases} y_n &= x_n - u_n, \\ z_n &= y_n - \dfrac{f(y_n)}{f'_n}\dfrac{1}{[1-r_n]^2}, \\ x_{n+1} &= z_n - \dfrac{f(z_n)}{f'_n}\dfrac{1}{[1-H(r_n)J(t_n)P(v_n)]^2}, \end{cases} \tag{13}$$

where

$$r_n = \frac{f(y_n)}{f_n}, \tag{14}$$

$$t_n = \frac{f(z_n)}{f_n}, \tag{15}$$

$$v_n = \frac{f(z_n)}{f(y_n)}, \tag{16}$$

and $H(r)$, $J(t)$, $P(v)$ are real-valued weight functions satisfying:

$$\begin{aligned} &H(0)J(0)P(0) = 2, \\ &H'(0)P(0)J(0) = -1, \\ &H''(0)P(0)J(0) = -1, \\ &H'''(0)P(0)J(0) = 3, \\ &|H^{(4)}(0)| < \infty, \\ &J'(0) = -3J(0)/8, \\ &|J''(0)| < \infty, \\ &P'(0) = -P(0)/4, \\ &|P''(0)| < \infty. \end{aligned}$$

Chun and Neta [20] have considered rational polynomials for each of the weight functions:

$$\begin{aligned} H(r) &= \frac{a + br + cr^2}{1 + dr + gr^2}, \\ J(t) &= \frac{\alpha + \beta t}{1 + \gamma t}, \\ P(v) &= \frac{A + Bv}{1 + Cv}. \end{aligned}$$

Since we only use the product $H(r)J(t)P(v)$, it is easy to see that

$$\begin{aligned} H(r) &= \frac{1}{2}\frac{4 + (2 - 8g)r + (8g - 3)r^2}{1 + (1 - 2g)r + gr^2}, \\ J(t) &= \frac{1}{8}\frac{8 + (8\gamma - 3)t}{1 + \gamma t}, \\ P(v) &= \frac{1}{4}\frac{4 + (4C - 1)v}{1 + Cv}. \end{aligned}$$

Thus, we have the three free parameters $g,\ \gamma$, and C.

We will consider the best four cases (denoted CN8a, CN8b, CN8c, and CN8d, respectively) for the three parameters:

(a) $g = -4,\ \ \gamma = 0,\ \ C = -4.$
(b) $g = -4,\ \ \gamma = 0,\ \ C = 0.$
(c) $g = 0,\ \ \gamma = 0,\ \ C = 0.$
(d) $g = 0,\ \ \gamma = 0,\ \ C = -4.$

(ix) A weight function-based eighth-order (WM8) optimal method [2] using the fourth-order Maheshwari's method ([30]) is given by

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= x_n - \left[r_n^2 - \frac{1}{r_n - 1}\right]u_n, \\ x_{n+1} &= z_n - \left[\phi(r_n) + \frac{t_n}{r_n - at_n} + 4t_n\right]\frac{f(z_n)}{f'_n}, \end{aligned} \tag{17}$$

where ϕ is an arbitrary real function satisfying the conditions

$$\phi(0) = 1,\ \phi'(0) = 2,\ \phi''(0) = 4,\ \phi'''(0) = -6. \tag{18}$$

Chun and Neta [31] have shown that this family cannot compete with WL8, and we will not experiment with it here.

(x) Thukral-Petković optimal eighth-order method [32]

$$
\begin{aligned}
y_n &= x_n - u_n,\\
z_n &= y_n - \frac{f(y_n)}{f_n'}\left[\frac{1+\beta r_n}{1+(\beta-2)r_n}\right],\\
x_{n+1} &= z_n - \frac{f(z_n)}{f_n'}\left[\phi(r_n) + \frac{t_n}{r_n - a t_n} + 4t_n\right],
\end{aligned}
\tag{19}
$$

where r_n is given by (14) and $\phi(r)$ is a real-valued weight function satisfying the conditions (to ensure eighth-order convergence)

$$
\phi(0) = 1,\ \phi'(0) = 2,\ \phi''(0) = 10-4\beta,\ \phi'''(0) = 12\beta^2-72\beta+72. \tag{20}
$$

Chun and Neta [21] have shown that this family cannot compete with WL8, and we will not experiment with it here.

(xi) Bi-Ren-Wu's optimal eighth-order method [33]

$$
\begin{aligned}
y_n &= x_n - u_n,\\
z_n &= y_n - \frac{f(y_n)}{f_n'}\frac{1-\frac{1}{2}r_n}{1-\frac{5}{2}r_n},\\
x_{n+1} &= z_n - p(t_n)\frac{f(z_n)}{f[z_n, y_n] + f[z_n, x_n, x_n](z_n - y_n)},
\end{aligned}
\tag{21}
$$

where t_n is given by (15) and the weight function $p(t)$ should satisfy the following condition to guarantee eighth order:

$$
p(0) = 1,\ \ p'(0) = 2. \tag{22}
$$

Bi et al. [33] have used

$$
p(t) = \frac{1}{(1-\alpha t)^{2/\alpha}}, \tag{23}
$$

with α, a non-zero real number, chosen as unity. In [16], we have considered the more general weight function

$$
p(t) = \frac{a+bt}{1+ct+gt^2}, \tag{24}
$$

satisfying the condition (22) and have shown that this family cannot compete with WL8, and we will not experiment with it here.

The next four methods were analyzed and compared to WL8 in [15] and found that they cannot compete with WL8. Therefore, we will not show those results here.

(xii) Lotfi et al.'s optimal family of methods [34]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - u_n \frac{r_n}{1-2r_n}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)} \frac{H(r_n)+K(t_n)}{G(v_n)}, \end{aligned} \tag{25}$$

where r_n is given by (14), t_n is given by (15), and v_n is given by (16). The weight functions $H,\ K,\ G$ satisfy

$$G(0) = 1,\ G'(0) = -1, \tag{26}$$

$$K(0) = 0,\ K'(0) = 2, \tag{27}$$

$$H(0) = 1,\ H'(0) = 2,\ H''(0) = 10,\ H'''(0) = 72. \tag{28}$$

(xiii) BCST, a method by Babajee et al. [35]

$$\begin{aligned} y_n &= x_n - u_n(1+u_n^5), \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}(1-r_n)^{-2}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)} \frac{1+r_n^2+5r_n^4+v_n}{(1-r_n-t_n)^2}, \end{aligned} \tag{29}$$

where $r_n,\ t_n$ and v_n are given by (14)–(16).

(xiv) CFGT, a method by Cordero et al. [36]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)} \frac{1}{1-2r_n-r_n^2-r_n^3/2}, \\ x_{n+1} &= z_n - \frac{1+3t_n}{1+t_n} \frac{f(z_n)}{f[z_n,y_n]+f[z_n,x_n,x_n](z_n-y_n)}, \end{aligned} \tag{30}$$

where r_n and t_n are given by (14)–(15) and the divided differences

$$f[z_n, y_n] = \frac{f(z_n)-f(y_n)}{z_n-y_n},$$

and

$$f[z_n, x_n, x_n] = \frac{f[z_n,x_n]-f'(x_n)}{z_n-x_n}.$$

Remark: In the third substep, the second term on the right is similar to King's method correction with $\beta = 3$. It is possible to use other values of β without affecting the order.

(xv) WL8-2, a family of methods by Wang and Liu [37]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= x_n - u_n G(r_n), \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)}(H(r_n)+V(r_n)W(v_n)), \end{aligned} \tag{31}$$

where r_n and v_n are given by (14) and (16), respectively, and the weight functions are

$$
\begin{aligned}
G(r_n) &= \frac{1-r_n}{1-2r_n},\\
H(r_n) &= \frac{5-2r_n+r_n^2}{5-12r_n},\\
V(r_n) &= 1+4r_n,\\
W(v_n) &= v_n.
\end{aligned} \tag{32}
$$

The next three methods were analyzed and compared to WL8 in [16] and found that they cannot compete with WL8. Therefore, we will not show those results here.

(xvi) Sharma and Sharma's optimal family of methods [38]

$$
\begin{aligned}
y_n &= x_n - u_n,\\
z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{1-2r_n},\\
x_{n+1} &= z_n - W(t_n)\frac{f(z_n)f[x_n,y_n]}{f[x_n,z_n]f[y_n,z_n]},
\end{aligned} \tag{33}
$$

with weight function

$$
W(t_n) = 1 + \frac{t_n}{1+\alpha t_n}, \tag{34}
$$

where t_n is given by (15) and α is some real parameter. Sharma and Sharma [38] have used $\alpha = 1$.

(xvii) Cordero et al.'s optimal family of methods [39]

$$
\begin{aligned}
y_n &= x_n - u_n,\\
z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{1-2r_n},\\
x_{n+1} &= \omega_n - \frac{f(z_n)}{f'(x_n)}\frac{3(\beta_2+\beta_3)(\omega_n-z_n)}{\beta_1(\omega_n-z_n)+\beta_2(y_n-x_n)+\beta_3(z_n-x_n)},
\end{aligned} \tag{35}
$$

where

$$
\omega_n = z_n - \frac{f(z_n)}{f'(x_n)}\left(\frac{1-r_n}{1-2r_n}+\frac{1}{2}\frac{v_n}{1-2v_n}\right)^2, \tag{36}
$$

where r_n and v_n are given by (14) and (16), respectively, and β_1, β_2, and β_3 are real parameters with $\beta_2 + \beta_3 \neq 0$.

Remark: Cordero et al. [39] have used $\beta_1 = \beta_3 = 0$ and $\beta_2 = 1$.

(xviii) Chun and Lee's optimal family of methods [40]

$$
\begin{aligned}
y_n &= x_n - u_n,\\
z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{(1-r_n)^2},\\
x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)}\frac{1}{(1-H(r_n)-J(t_n)-P(v_n))^2},
\end{aligned} \tag{37}
$$

where r_n, t_n, and v_n are given by (14)–(16), and the weight functions should satisfy the following conditions to guarantee eighth order:

$$H(0) = 0,\ H'(0) = 1,\ H''(0) = 1,\ H'''(0) = -3, \tag{38}$$

$$J(0) = 0,\ J'(0) = \frac{1}{2},\ \ P(0) = 0,\ P'(0) = \frac{1}{2}. \tag{39}$$

Remark: Chun and Lee [40] have used the following weight functions

$$\begin{aligned} &H(r_n) = -\beta - \gamma + r_n + r_n^2/2 - r_n^3/2,\\ &J(t_n) = \beta + t_n/2,\\ &P(v_n) = \gamma + v_n/2, \end{aligned} \tag{40}$$

and β and γ are real parameters chosen to be zero for simplicity.

In our previous work, we found that it is better not to use polynomials as weight functions, therefore we will use the following:

$$\begin{aligned} J(t) &= \frac{a_1 + b_1 t}{1 + \delta_1 t},\\ P(t) &= \frac{a_2 + b_2 t}{1 + \delta_2 t},\\ H(t) &= \frac{a_3 + b_3 t + c_3 t^2}{1 + \delta_3 t + g_3 t^2}. \end{aligned} \tag{41}$$

These functions satisfying the conditions (38)–(39) are given by

$$J(t) = \frac{1}{2}\frac{t}{1 + \delta_1 t}, \tag{42}$$

$$P(t) = \frac{1}{2}\frac{t}{1 + \delta_2 t}, \tag{43}$$

$$H(t) = \frac{1}{2}\frac{2t + (3 - 4g_3)t^2}{1 + (1 - 2g_3)t + g_3 t^2}. \tag{44}$$

(xix) Bi-Wu-Ren's another optimal eighth-order (BWR8) method [41]

$$\begin{aligned} y_n &= x_n - u_n,\\ z_n &= y_n - \frac{f(y_n)}{f'_n} h(r_n),\\ x_{n+1} &= z_n - \frac{f_n + \beta f(z_n)}{f_n + (\beta - 2) f(z_n)} \cdot \frac{f(z_n)}{f[z_n, y_n] + f[z_n, x_n, x_n](z_n - y_n)}, \end{aligned} \tag{45}$$

where r_n is given by (14) and h is a weight function satisfying

$$h(0) = 1,\quad h'(0) = 2,\quad h''(0) = 10,\quad |h'''(0)| < \infty. \tag{46}$$

We have looked at the four special cases presented in [41] and noticed that the method denoted G81 there is the best performer. In this case, $\beta = 3$ and the weight function

$$h(t) = \frac{1 - t/2}{1 - 5t/2}.$$

We will call it BWR8 and experiment with it.

(xx) Khan et al.'s optimal eighth-order family of methods (KFS8) [42]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'} \frac{1}{1 - 2r_n + \omega_1 r_n^2}, \\ x_{n+1} &= z_n - \frac{1}{1 + \omega_2 t_n^2} \cdot \frac{f(z_n)}{f[z_n, y_n] - C(y_n - z_n) - D(y_n - z_n)^2}, \end{aligned} \tag{47}$$

where

$$D = \frac{f_n' - f[x_n, y_n]}{(x_n - y_n)(x_n - z_n)} - \frac{f[x_n, y_n] - f[y_n, z_n]}{(x_n - z_n)^2}, \tag{48}$$

and

$$C = \frac{f[x_n, y_n] - f[y_n, z_n]}{x_n - z_n} - D(x_n + y_n - 2z_n). \tag{49}$$

Sharma and Arora [44] have used $\omega_1 = \omega_2 = 1$ in their comparison.

(xxi) Kou et al.'s optimal eighth-order family of methods

The previous family can be considered a special case of the family in Kou et al. [43]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{\psi(x_n, y_n)}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{\Gamma(x_n, y_n, z_n)}, \end{aligned} \tag{50}$$

where $\psi(x_n, y_n)$ is a real function using the evaluation of f_n, f_n', and $f(y_n)$ and where $\Gamma(x_n, y_n, z_n)$ is a real function using the evaluation of $f(z_n)$ as well as f_n, f_n', and $f(y_n)$. Kou et al. have suggested to use,

$$\psi(x_n, y_n) = \frac{f_n + (\beta - 2) f(y_n)}{f_n + \beta f(y_n)} f_n' \tag{51}$$

and $\Gamma(x_n, y_n, z_n)$ is computed by using a cubic interpolating polynomial, so that the method (50) is eighth order. This suggests $\Gamma(x_n, y_n, z_n) = \frac{b}{\Phi(x_n, y_n, z_n)}$ with $b = f[z_n, y_n] - C(y_n - z_n) - D(y_n - z_n)^2$ and C and D are given by (49) and (48), respectively. Kou et al. suggested two possibilities for $\Phi(x_n, y_n, z_n)$, denoted KWL81 and KWL82. Khan et al. used another choice for $\Gamma(x_n, y_n, z_n)$ and a different choice for $\psi(x_n, y_n)$.

(xxii) Džunić and Petković eighth-order (DP8) method, see (4.93) on page 152 of [2]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{f(y_n)}{f_n'} \frac{1}{1 - 2r_n}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f_n'} \cdot \frac{1}{(1 - 2r_n - r_n^2)(1 - v_n)(1 - 2t_n)}, \end{aligned} \tag{52}$$

(xxiii) Sharma and Arora's optimal eighth-order (SA8) method [44]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= \phi_4(x_n, y_n), \\ x_{n+1} &= z_n - \frac{f[z_n, y_n]}{f[z_n, x_n]} \frac{f(z_n)}{2f[z_n, y_n] - f[z_n, x_n]}, \end{aligned} \tag{53}$$

where

$$\phi_4(x_n, y_n) = y_n - \frac{f(y_n)}{2f[y_n, x_n] - f'_n}. \tag{54}$$

The method (53) is of eighth order when $\phi_4(x_n, y_n)$ is replaced with any two-point optimal fourth-order method [44]. Sharma and Arora have considered three special cases for $\phi_4(x_n, y_n)$ and have demonstrated that the case based on Ostrowski's iteration given by (54) performs best, so we will experiment with this here. It is easy to show that (53) with (54) can be written as

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{1}{1 - 2r_n} \cdot \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} &= z_n - q(r_n, v_n) \frac{f(z_n)}{f'(x_n)}, \end{aligned} \tag{55}$$

where

$$q(r_n, v_n) = \frac{4r_n^2(1 - r_n)^2}{(1 - 2r_n)} \cdot \frac{(1 - v_n)}{[(1 - 2r_n)^2 + (3 - 4r_n)r_n v_n](-1 + r_n v_n)}. \tag{56}$$

(xxv) Cordero-Torregrosa-Vassileva's (CTV8) optimal method [45]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= x_n - \frac{1 - r_n}{1 - 2r_n} u_n, \\ x_{n+1} &= z_n - \left(\frac{1 - r_n}{1 - 2r_n} + \frac{v_n}{a_2 m_2 v_n - 1} \right)^2 \frac{n_1 + n_2 v_n}{n_1 + (n_2 - 3n_1) v_n} \frac{f(z_n)}{f'(x_n)}, \end{aligned} \tag{57}$$

where $n_1,\ n_2,\ a_2$, and m_2 are real parameters with $m_2 \neq 0$ and $n_1 \neq 0$.

In their experiments, they have used $m_2 = n_1 = 1$ and $a_2 = n_2 = 0$.

(xxv) Džunić-Petković-Petković's optimal method (DPP8) [46]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - p(r_n) \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} &= z_n - q(r_n, v_n) \frac{f(z_n)}{f'(x_n)}, \end{aligned} \tag{58}$$

where $p(0) = 1$, $p'(0) = 2$, $p''(0) = 4$, $q(0,0) = 1$, $q_r(0,0) = 2$, $q_v(0,0) = 1$, $q_{rr}(0,0) = 6$, $q_{rv}(0,0) = 4$. We will experiment with one of the members they used, namely

$$p(r_n) = \frac{1 + r_n + r_n^2}{1 - r_n + r_n^2},$$

$$q(r_n, v_n) = \frac{1 - 4r_n + v_n}{(1 - 3r_n)^2 + 2r_n v_n}.$$

It is clear that (55) and (57) are special cases of (58).

(xxvi) Geum and Kim's optimal method (GK8) [47]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - p(r_n)\frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} &= z_n - q(r_n, v_n)\frac{f(z_n)}{f'(x_n)}, \end{aligned} \tag{59}$$

This is the same as (58), but the authors here have taken

$$p(r_n) = \frac{1 + \beta r_n + \lambda r_n^2}{1 + (\beta - 2)r_n + \mu r_n^2}$$

and

$$q(r_n, v_n) = \frac{1}{1 - 2r_n - v_n}$$

where $\mu = -3\beta/2$ and $\lambda = -1 + \beta/2$. The authors experimented with three values of the parameter $\beta = 0, 2, -4/3$.

(xxvii) Liu and Wang's optimal method (LW8) [48]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{1}{1 - 2r_n}\frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} &= z_n - \left[\left(\frac{1 - r_n}{1 - 2r_n}\right)^2 + \frac{v_n}{1 - \alpha v_n} + G(t_n)\right]\frac{f(z_n)}{f'(x_n)}, \end{aligned} \tag{60}$$

where $G(0) = 0$, $G'(0) = 4$. They have experimented with three members and we will pick the one with $\alpha = 5$ for which

$$G(t_n) = \frac{4t_n}{1 - 7t_n}.$$

Note that $t_n = r_n v_n$ and thus this family is also included in (58).

(xxviii) Sharma and Arora weighted Newton optimal method (SAWN8) [49]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= \phi_4(x_n, y_n), \\ x_{n+1} &= z_n - \frac{f'(x_n) - f[y_n, x_n] + f[z_n, y_n]}{2f[z_n, y_n] - f[z_n, x_n]}\frac{f(z_n)}{f'(x_n)}. \end{aligned} \tag{61}$$

We will experiment with one of the three members suggested by the authors, namely when

$$\phi_4(x_n, y_n) = y_n - \frac{f(y_n)}{2f[y_n, x_n] - f'(x_n)}.$$

(xxix) Sharma-Guha-Gupta's optimal method (SGG8) [50]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= y_n - \frac{1 + ar_n}{1 + (a-2)r_n} \frac{f(y_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \frac{P + Q + R}{Pf[z_n, x_n] + Qf'(x_n) + Rf[y_n, x_n]} f(x_n), \end{aligned} \tag{62}$$

where

$$\begin{aligned} P &= (x_n - y_n) f(x_n) f(y_n), \\ Q &= (y_n - z_n) f(y_n) f(z_n), \\ R &= (z_n - x_n) f(z_n) f(x_n). \end{aligned} \tag{63}$$

They used $a = 0$ in their experiments.

(xxx) Thukral's optimal method (T8) [51]

$$\begin{aligned} y_n &= x_n - u_n, \\ z_n &= x_n - \frac{1 + r_n^2}{1 - r_n} u_n = y_n - \frac{r_n(1 + r_n)}{1 - r_n} u_n, \\ x_{n+1} &= z_n - \left[\left(\frac{1 + r_n^2}{1 - r_n} \right)^2 - 2r_n^2 - 6r_n^3 + v_n + 4t_n \right] \frac{f(z_n)}{f'(x_n)}. \end{aligned} \tag{64}$$

2 Extraneous fixed points

In this section, we introduce the notion of extraneous fixed points and show how to find those for any given method. It is easy to see that any method can be written as

$$x_{n+1} = x_n - H_f \frac{f_n}{f'_n}, \tag{65}$$

where the function H_f depends on x_n and other intermediate values. In Tables 1 and 2, we list the function H_f for each of the methods discussed here.

It is clear that if x_n is a zero of the function $f(x)$ then x_n is a fixed point of the iterative method (65). But even if x_n is a zero of H_f and not of $f(x)$, it is a fixed point. Those fixed points that are zeroes of H_f and not of $f(x)$ are called extraneous fixed points. For example, Newton method does not have any extraneous fixed point, since $H_f = 1$. In order to find the extraneous fixed points, we substitute the quadratic polynomial $z^2 - 1$ for $f(z)$ and then find the zeros of H_f. For example, Super Halley method has extraneous fixed points which are the solution of $L_f = 2$,which are (see [12]) $\pm\frac{\sqrt{3}}{3}i$.

Table 1 The function H_f for each of the methods

Method	H_f
NJ8	$\dfrac{f_n'}{\frac{1}{6}f_n' + \frac{1}{6}f'(y_n) + \frac{2}{3}f'(\eta_n)} + t_n \dfrac{f_n' + f'(y_n) + a_2 f'(\eta_n)}{(-1-a_2)f_n' + (3+a_2)f'(y_n) + a_2 f'(\eta_n)}$
NP8	$1 - c_n f_n f_n' + d_n f_n^2 f_n'$
N8	$1 - \gamma f_n f_n' + \rho f_n^2 f_n'$
HK8	$1 + r_n \dfrac{1+\beta r_n}{1+(\beta-2)r_n} + \dfrac{H_3(z_n)}{f_n}\dfrac{f_n'}{f'(z_n)}$
HKT8	$1 + r_n \dfrac{1}{(1-r_n)^2} + t_n \dfrac{f_n'}{H_3'(z_n)}$
KT8	$1 + r_n[1-r_n]^2 + \dfrac{r_n f(z_n)}{[1-r_n]^2}\dfrac{1+r_n[r_n-t_n]}{[1-t_n]^2[f(y_n)-f(z_n)]}$
WL8	$1 + \dfrac{r_n}{1-2r_n} + \dfrac{f_n'}{H_3'(z_n)}t_n$
CN8	$1 + \dfrac{r_n}{[1-r_n]^2} + \dfrac{t_n}{[1-H(r_n)J(t_n)P(v_n)]^2}$
BWR8	$1 + r_n h(r_n) + \dfrac{f_n'}{f_n}\dfrac{1+\beta t_n}{1+(\beta-2)t_n}\dfrac{f(z_n)}{f[z_n,y_n]+f[z_n,x_n,x_n](z_n-y_n)}$
KFS8	$1 + \dfrac{r_n}{1-2r_n+\omega_1 r_n^2} + \dfrac{t_n}{1+\omega_2 t_n^2}\cdot\dfrac{f_n'}{f[z_n,y_n]-C(y_n-z_n)-D(y_n-z_n)^2}$
KWL81	$1 + r_n \dfrac{1+\beta r_n}{1+(\beta-2)r_n} + \dfrac{t_n f_n'}{b}$
KWL82	$1 + r_n \dfrac{1+\beta r_n}{1+(\beta-2)r_n} + \dfrac{t_n f_n'}{b}\cdot\left[1+\dfrac{Cf(z_n)}{b^2-\alpha C f(z_n)}\right]$
DP8	$1 + \dfrac{r_n}{1-2r_n} + \dfrac{t_n}{(1-2r_n-r_n^2)(1-v_n)(1-2t_n)}$
SA8	$1 + \dfrac{f_n' r_n}{2f[y_n,x_n]-f_n'} + \dfrac{f_n'}{f_n}\dfrac{f[z_n,y_n]}{f[z_n,x_n]}\cdot\dfrac{f(z_n)}{2f[z_n,y_n]-f[z_n,x_n]}$

In our previous work, we found that methods without extraneous fixed point or those having such points on the imaginary axis perform better than others. For families of methods, we showed how to choose the parameter(s) such that the extraneous fixed points are on or close to the imaginary axis. When a method contains a weight function, we suggested a rational function as a weight function. This leading to a family of methods with at least one parameter. We also demonstrated that a polynomial weight function does not give as good results.

To choose the parameters in the methods, the following criterion can be used, which was developed in [21] and is defined below.

Let $E = \{z_1, z_2, ..., z_n\}$ be the set of the extraneous fixed points corresponding to the values given to the parameters. We define

$$d = \max_{z_i \in E} |Re(z_i)|. \tag{66}$$

Table 2 The function H_f for each of the methods (continued)

Method	H_f
CTV8	$\frac{1-r_n}{1-2r_n}+\left(\frac{1-r_n}{1-2r_n}+\frac{v_n}{a_2m_2v_n-1}\right)^2\frac{t_n}{1-3v_n}$
DPP8	$1+r_n\frac{1+r_n+r_n^2}{1-r_n+r_n^2}+t_n\frac{1-4r_n+v_n}{(1-3r_n)^2+2t_n}$
GK8	$1+r_n\frac{1+\beta r_n+\lambda r_n^2}{1+(\beta-2)r_n+\mu r_n^2}+\frac{t_n}{1-2r_n-v_n}$
LW8	$1+\frac{r_n}{1-2r_n}+\left[\left(\frac{1-r_n}{1-2r_n}\right)^2+\frac{v_n}{1-5v_n}+\frac{4t_n}{1-7t_n}\right]t_n$
SAWN8	$1+\frac{r_n}{\frac{2f[y_n,x_n]}{f'(x_n)}-1}+\frac{f'(x_n)-f[y_n,x_n]+f[z_n,y_n]}{2f[z_n,y_n]-f[z_n,x_n]}t_n$
SGG8	$\frac{(P+Q+R)f'(x_n)}{Pf[z_n,x_n]+Qf'(x_n)+Rf[y_n,x_n]}$
T8	$\frac{1+r_n^2}{1-r_n}+\left[\left(\frac{1+r_n^2}{1-r_n}\right)^2-2r_n^2-6r_n^3+v_n+4t_n\right]t_n$

We look for the parameters which attain the minimum of the function d given in (66).

We now quote the results obtained previously for each of the methods. For the method N8, we have used $\beta = 2$ for simplicity. We have also taken $\beta = -0.53$ (denoted N8d).

For HK8, we have previously used $\beta = 3 - 2\sqrt{2}$ which is the optimal parameter for King's method (see [12]). Here, we have taken $\beta = 0$ (denoted HK8d). There are other values of β leading to purely imaginary extraneous fixed points, but $\beta = 0$ has the least number of such points (7 vs 10).

For WL8, we found that the method has six purely imaginary extraneous fixed points, $\pm 2.07652139657i$, $\pm 0.797473388882i$, $\pm 0.228243474390i$. All fixed points are repulsive. In our previous paper [13], we made a mistake saying that WL8 has no extraneous fixed points.

For KFS8, we found that the extraneous fixed points are purely imaginary when $\omega_1 = \omega_2 = 0$. This means that the additional factor in the third substep is unity and therefore the method is identical to the one suggested by Kou et al.

For KWL81, the case when $\Phi(x_n, y_n, z_n) = 1$, we have purely imaginary fixed points only when $\beta = 0$. This is identical to KFS8. Therefore, we do not need to show KFS8 here. For KWL82, the case that $\Phi(x_n, y_n, z_n) = 1 + \frac{Cf(z_n)}{b^2-\alpha Cf(z_n)}$, the extraneous fixed point are purely imaginary when $\beta = 0$ and α is in the interval $[1, 3]$. We will present only the case $\beta = 0$, $\alpha = 2$ which we denote KWL82a2. The reason is that other values of α lead to inferior results.

The extraneous fixed points for KWL81 are: $\pm 2.07652139657i$, $\pm 0.797473388882i$, and $\pm 0.228243474390i$. All extraneous fixed points are repulsive. Note that these are the same as the extraneous fixed points for WL8. In fact, we can show that the two methods are identical. The numerical results in the tables are not identical because of the way the iterates are computed.

The extraneous fixed points for KWL82a2 are $\pm 4.70463011i$, $\pm 2.24603677i$, $\pm 1.37638192i$, $\pm 0.900404044i$, $\pm 0.577350269i$, $\pm 0.324919696i$, and $\pm 0.105104235i$. All extraneous fixed points are repulsive.

For SA8, we found that the method has eight purely imaginary extraneous fixed points, $\pm 2.74747741945462i$, $\pm 1.19175359259421i$, $\pm 0.5773502692i$, $\pm 0.176326980708465i$. All extraneous fixed points are repulsive.

The extraneous fixed points for CTV8 are $\pm 0.310198439929491 \pm 0.971937369115815i$, ± 0.169642214518236, ± 3.25348840711669i, $\pm 1.180338081i$, $\pm 0.4858509501i$, and $\pm 0.2759381566i$. All extraneous fixed points are repulsive.

The extraneous fixed points for DPP8 are $\pm 1.15744508259162 \pm 0.975896785413439i$, $\pm 0.483003623093895 \pm 0.529644104091676i$, $\pm 0.417259030165903 \pm 0.354453089732201i$, $\pm 0.390469097431474 \pm 0.242913168967612i$, $\pm 0.366915096535168 \pm 0.397205750373932i$, $\pm 0.184782116230004 \pm 0.107079573283538i$, and ± 0.305847098351993. All extraneous fixed points are repulsive.

For GK8 when $\beta = 0$, we have $\pm 0.229435172737268 \pm 0.770167980885006i$, $\pm 0.193629632701682 \pm 0.303836169651621i$, $\pm 2.41371611097065i$, $\pm 0.6821976662i$, ± 0.400870978608947 and when $\beta = 2$

± 0.570582062587072, ± 0.547435187983705, ± 0.540696888276458, ± 0.308077251841470, $\pm 0.658552579276995i$, $\pm 2.41868237788682i$, $\pm 0.310587859633575i$ and when $\beta = -4/3$

$\pm 0.316170911320087 \pm 0.244928706792940i$, $\pm 0.292265018108880 \pm 0.468142542228367i$, $\pm 0.161858034057866 \pm 1.15243503420192i$, $\pm 2.41315986399038i$, $\pm 1.149218897i$, ± 0.337486852528835. All extraneous fixed points are repulsive for all the values of β.

The extraneous fixed points for LW8 are $\pm 0.443428970837523 \pm 0.404034320627720i$, $\pm 0.430878047225417 \pm 1.12786914351688i$, $\pm 0.208822949226219 \pm 0.339321998846708i$, $\pm 2.391136235i$, $\pm 0.7836279605i$, ± 4.29516917647657, ± 0.298401332673525. All extraneous fixed points are repulsive.

The extraneous fixed points for SAWN8 are $\pm 0.230237981705881 \pm 0.949019368568498i$, $\pm 2.70750917598407i$, $\pm 0.454012206979393i$, ± 0.158407505492566. All extraneous fixed points are repulsive.

For SGG8, we found that the method has six purely imaginary extraneous fixed points, $\pm 2.07652139657234i$, $\pm 0.797473388882404i$, $\pm 0.228243474390150i$. All fixed points are repulsive.

The extraneous fixed points for T8 are $\pm 0.560031644976194 \pm 0.915329728118228i$, $\pm 0.524296783378056 \pm 0.451348385219899i$, $\pm 0.481521406531424 \pm 0.179771147270053i$, $\pm 0.286425022277987 \pm 0.247206249464381i$, ± 0.471756447060788, $\pm 0.195796636991793i$. All extraneous fixed points are repulsive.

3 Numerical experiments

In this section, we detail the experiments we have used with each of the methods. For some methods, we have taken more than one case. All the examples have roots within a square of $[-3, 3]$ by $[-3, 3]$. We have taken 360,000 equally spaced points in the square as initial points for the methods and we have registered the total number of iterations required to converge to a root and also to which root it converged. We have also collected the CPU time (in seconds) required to run each method on all the points using Dell Optiplex 990 desktop computer. We then computed the average number of function evaluations required per point and the number of points requiring 40 iterations.

Example 1 The first example is the quadratic polynomial

$$p_1(z) = z^2 - 1 \tag{67}$$

whose roots are at ± 1.

The best results will be when the basins are divided by the imaginary axis. We have plotted the basins in Figs. 1 and 2. We used a different color for each basin, so that we can tell if the method converged to the closest root. We have also used lighter shade when the number of iterations is lower and at the maximum number of iterations we color the point black. Therefore ideally, the method should show lighter shades. The best methods are HK8d, WL8, KWL81, KWL82a2, SA8, and SGG8.

Now, we check Table 3 to see the average number of function evaluations per point. The minimum is 8.0 function evaluations per point on average, and it is achieved by KWL82a2 followed by methods: SA8 (8.65), KWL81 (9.04) and HK8d, WL8, GK8 with $\beta = -4/3$ and SGG8 with 9.06 and SAWN8 with 9.08 function evaluations per point on average. The highest number (36.82) was used by NP8. All other methods used 9.33–13.99 function evaluations per point on average. For this reason, we will not experiment with NP8 in the rest of the examples.

Based on the CPU time in seconds, we find that the fastest method is SA8 with 152.381 s. The slowest is NP8 with 1163.71 s. We can see that the basins for this method have many black points (Fig. 1, middle of top row). In terms of the number of black points (see Table 5), we find that most methods have 601 such points except NP8 (30825 points).

Example 2 The second example is the cubic polynomial

$$p_2(z) = z^3 - 1 \tag{68}$$

having the three roots of unity.

The basins of attraction are given in Figs. 3 and 4. Based on these plots, we find that HK8d, WL8, KWL81, KWL82a2, SA8, and SGG8 are best. Based on Table 3, we find that the minimum number of function evaluations per point is achieved by KWL82a2 (8.8) followed by SA8 (9.68). The worst are DPP8 (29.66), T8 (29.19), and BWR8 (18.19). All the other methods use 10.17–17.12.

The fastest method is SA8 method (224.969 s) and the slowest are T8 (743.937 s), DPP8 (678.355 s), BWR8 (618.7 s), KT8 (493.353 s), and N8 (466.022 s). Therefore, we will remove these five methods. Based on the number of black points, clearly we have BWR8 being the worst with 16,517 such points, T8 (13,994) and DPP8 (9729).

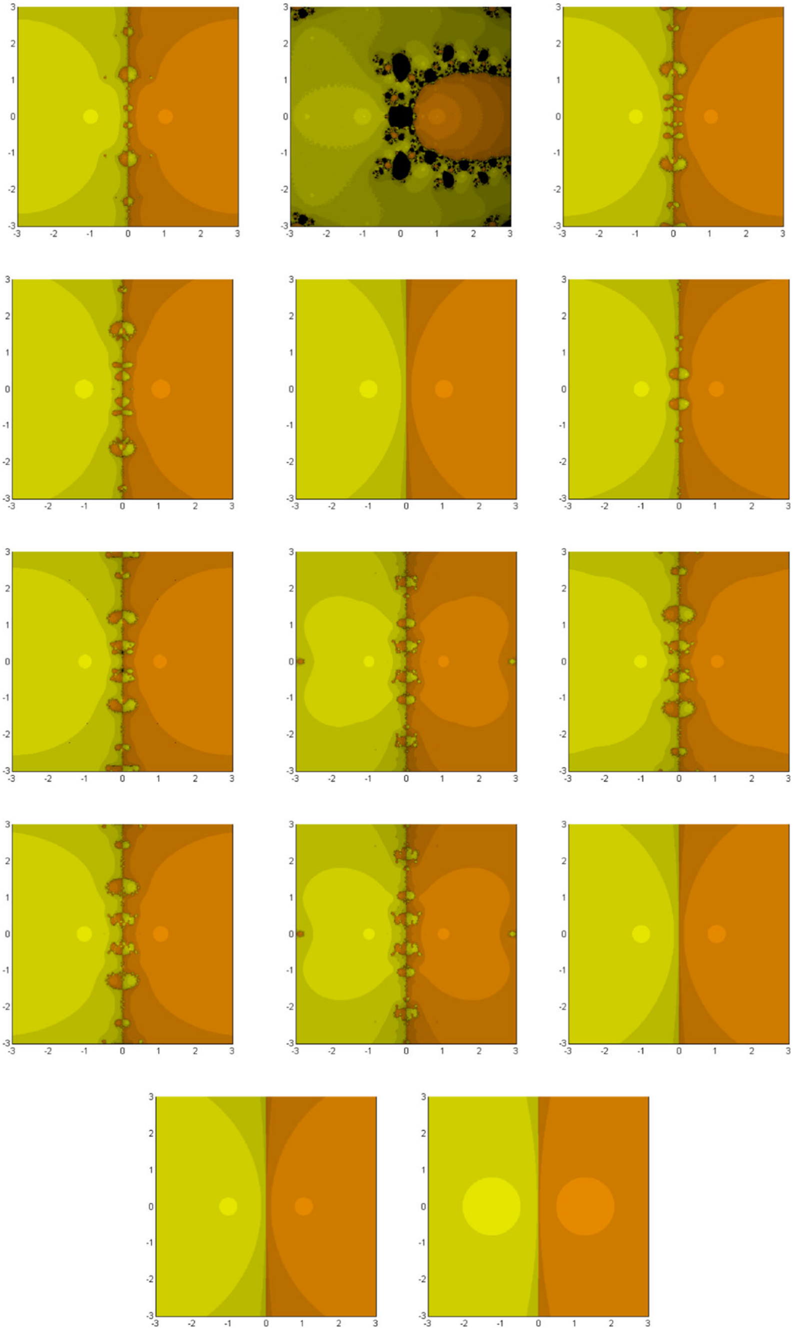

Fig. 1 The top row for NJ8 (*left*), NP8 (*center*), and N8 (*right*). The second row for N8d (*left*), HK8d (*center*), and HKT8 (*right*). The third row for KT8 (*left*), CN8a (*center*), and CN8b (*right*). The fourth row for CN8c (*left*), CN8d (*center*), and WL8 (*right*).The bottom row for KWL81 (*left*), KWL82a2 (*right*) for the roots of the polynomial $z^2 - 1$

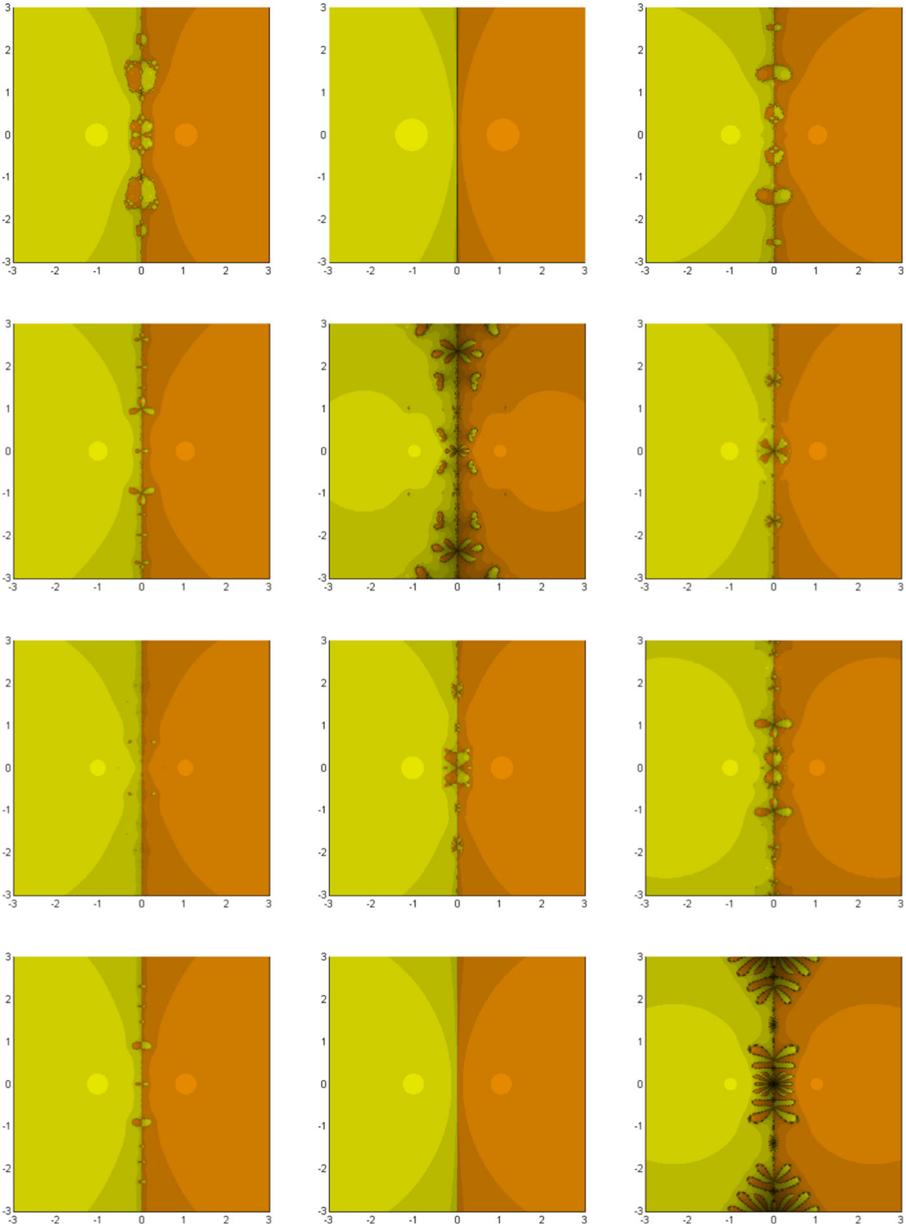

Fig. 2 The top row for BWR8 (*left*), SA8 (center), and DP8 (*right*). The second row for CTV8 (*left*), DPP8 (*center*), and GK with $\beta = 0$ (*right*). The third row for GK with $\beta = 2$ (*left*) and $\beta = -4/3$ (*center*) and LW8 (*right*). The last row for SAWN8 (*left*), SGG8 (*center*) and T8 (*right*) for the roots of the polynomial $z^2 - 1$

Table 3 Average number of function evaluations per point for each example (1–6) and each of the methods

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Average
NJ8	12.77	17.12	16.59	13.71	25.02	24.05	18.21
NP8	36.82	–	–	–	–	–	–
N8	10.22	13.34	–	–	–	–	–
N8d	9.63	12.45	12.32	–	–	–	-
HK8d	9.06	10.83	11.23	10.12	14.88	13.95	11.68
HKT8	9.53	12.59	12.04	10.28	18.57	16.73	13.29
KT8	10.60	14.11	–	–	–	–	–
WL8	9.06	10.83	11.23	9.45	16.17	13.74	11.75
CN8a	11.45	14.54	14.73	–	–	–	–
CN8b	10.57	13.83	13.52	–	–	–	–
CN8c	10.30	13.52	13.39	11.43	18.27	17.13	14.01
CN8d	11.41	14.55	14.78	–	–	–	–
KWL81	9.04	10.83	11.23	9.45	16.17	13.74	11.74
KWL82a2	8.0	8.8	9.36	9.44	14.60	13.04	10.54
BWR8	9.36	18.19	–	–	–	–	–
DP8	9.76	11.89	12.62	10.8	15.03	14.24	12.39
SA8	8.65	9.68	10.46	10.20	12.11	11.57	10.45
CTV8	9.36	10.76	11.19	10.87	15.73	13.89	11.97
DPP8	13.46	29.66	–	–	–	–	–
GK8 ($\beta = 0$)	9.33	12.01	–	–	–	–	–
GK8 ($\beta = 2$)	9.40	11.66	12.07	13.16	24.37	–	–
GK8 ($\beta = -4/3$)	9.06	11.87	–	–	–	–	–
LW8	10.23	13.88	13.84	12.21	20.99	18.81	14.99
SAWN8	9.08	10.66	11.00	10.77	15.13	13.09	11.62
SGG8	9.06	10.17	11.14	9.72	12.27	11.83	10.70
T8	13.99	29.19	–	–	–	–	–

We will also remove from consideration the two cases of GK8 with ($\beta = 0$ and $\beta = -4/3$), since the third one ($\beta = 2$) performed better in terms of the number of function evaluations per point on average.

Example 3 The third example is another cubic polynomial, but with real roots only, i.e., the polynomial is given by:

$$p_3(z) = z^3 - z. \tag{69}$$

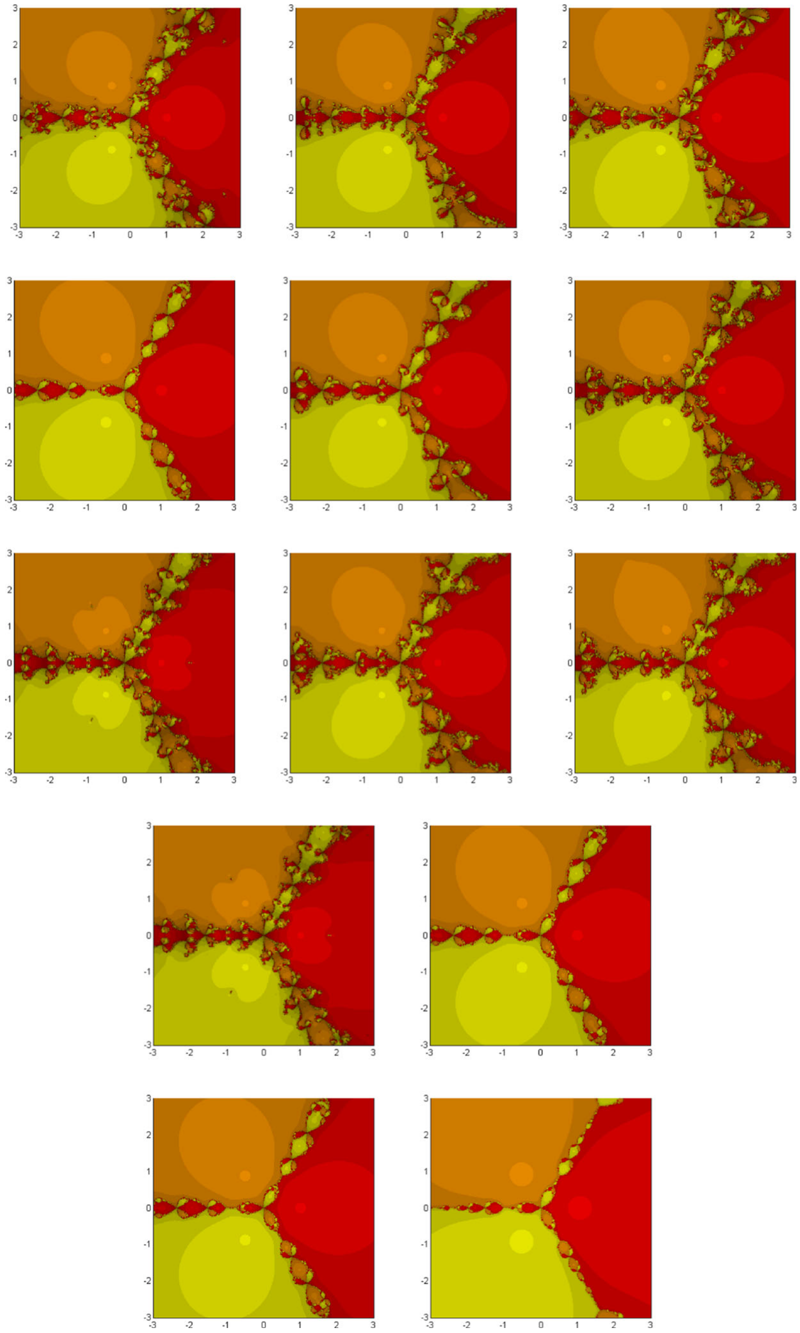

Fig. 3 The top row for NJ8 (*left*), N8 (*center*), and N8d (*right*). The second row for HK8d (*left*), HKT8 (*center*), and KT8 (*right*). The third row for CN8a (*left*), CN8b (*center*), and CN8c (*right*). The fourth row for CN8d (*left*) and WL8 (*center*). The bottom row for KWL81 (*left*), KWL82a2 (*right*) for the roots of the polynomial $z^3 - 1$

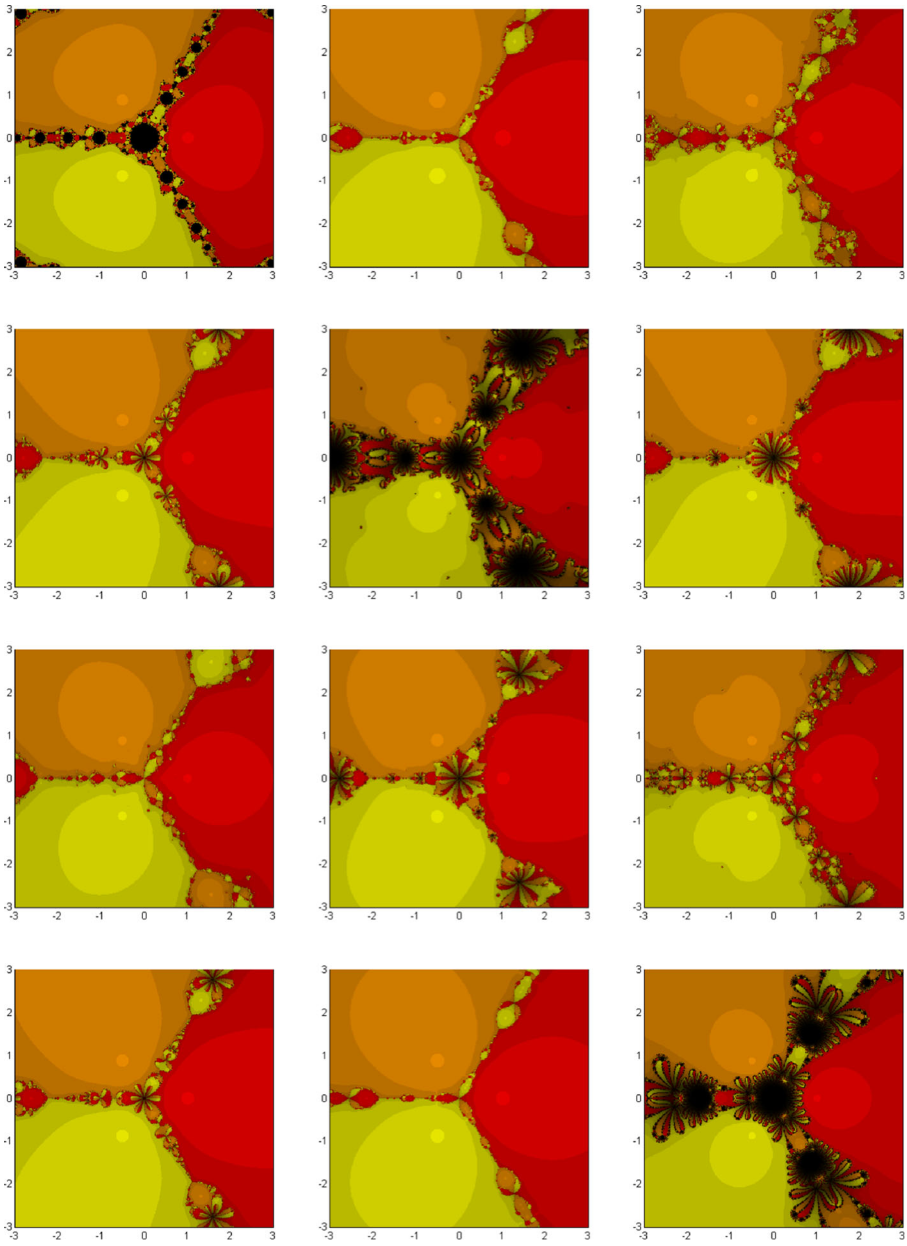

Fig. 4 The top row for BWR8 (*left*), SA8 (*center*), and DP8 (*right*). The second row for CTV8 (*left*), DPP8 (*center*), and GK with $\beta = 0$ (*right*). The third row for GK with $\beta = 2$ (*left*) and $\beta = -4/3$ (*center*) and LW8 (*right*). The last row for SAWN8 (*left*), SGG8 (*center*), and T8 (*right*) for the roots of the polynomial $z^3 - 1$

The basins of attraction are displayed in Figs. 5 and 6. It seems that the best methods are HK8d, WL8, KWL81, KWL82a2, SA8, and SGG8. Consulting the number

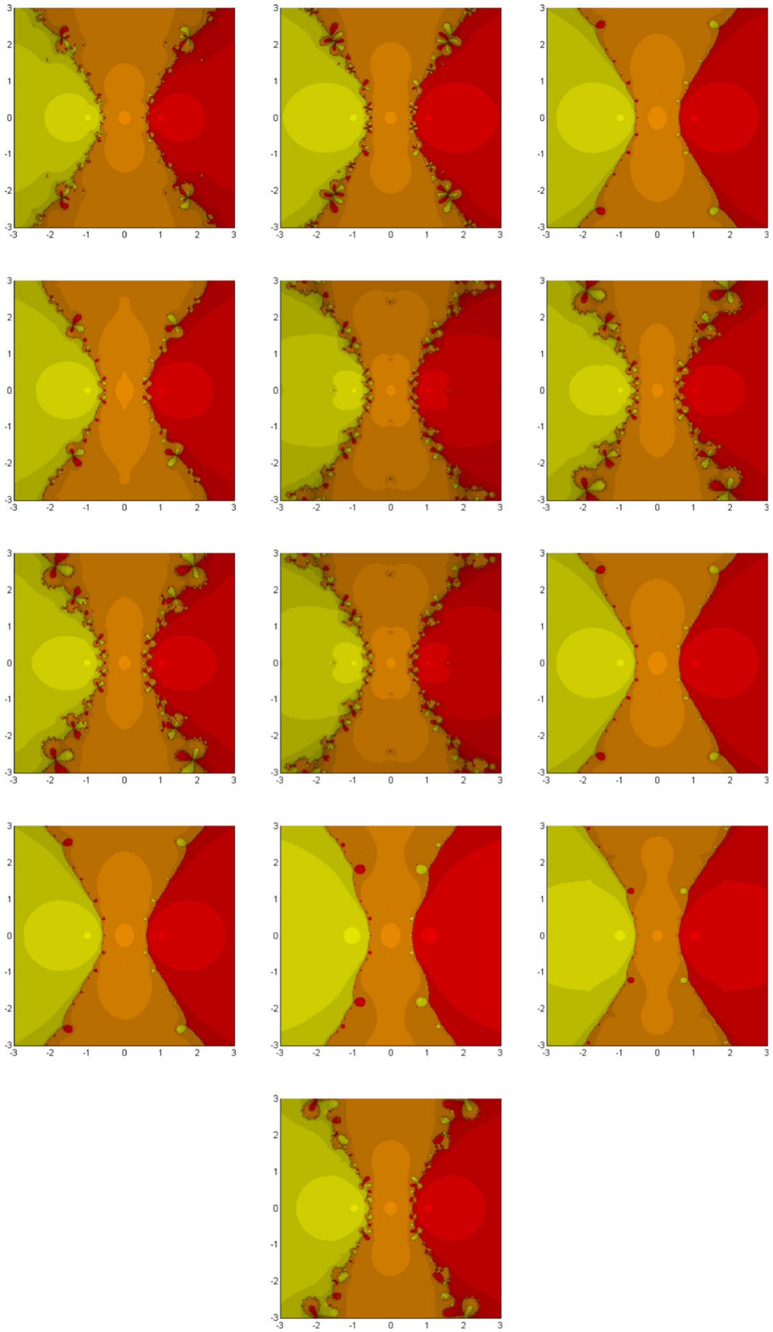

Fig. 5 The top row for NJ8 (*left*), N8d (*center*), and HK8d (*right*). The second row for HKT8 (*left*) and CN8a (*center*) and CN8b (*right*). The third row for CN8c (*left*), CN8d (*center*), WL8 (*right*). The fourth row for KWL81 (*left*), KWL82a2 (*center*), and SA8 (*right*). The bottom row for DP8 for the roots of the polynomial $z^3 - z$

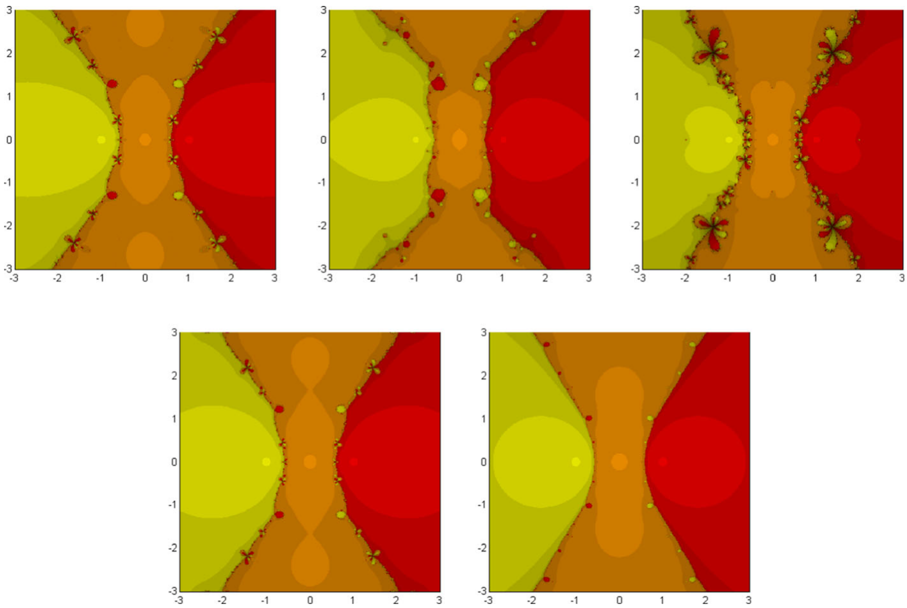

Fig. 6 The top row for CTV8 (*left*), GK with $\beta = 2$ (*center*), and LW8 (*right*). The bottom row for SAWN8 (*left*) and SGG8 (*right*) for the roots of the polynomial $z^3 - 1$

of function evaluations per point, we find that KWL82a2 is best (9.36) followed by SA8 (10.46), SAWN8 (11), SGG8 (11.14), CTV8 (11.19), HK8d, WL8, and KWL81 (11.23). The worst is NJ8 (16.59). All the others use 12.04–14.73 function evaluations per point. The fastest method is again SA8 (241.178 s). The slowest are CN8a (435.929), CN8d (419.33 s), CN8b (402.358), and N8d (398.036 s). We will remove these methods from further consideration. All method have no black points.

Example 4 The fourth example is a quartic polynomial with real roots at $\pm 1, \pm 3$.

$$p_4(z) = z^4 - 10z^2 + 9. \tag{70}$$

The basins are displayed in Fig. 7. The best methods are HK8d, WL8, KWL81, KWL82a2, SA8, and SGG8. Based on the average number of function evaluations per point (see Table 3), we find that the minimum is achieved by KWL82a2 (9.44), followed closely by KWL81 and WL8 (9.45), SGG8 (9.72), HK8d (10.12), and SA8 (10.20). The worst method in this sense is NJ8 which uses 13.71 function evaluation per point on average. The rest of the methods use 10.28–13.16 function evaluations per point on average. In terms of the CPU time, the fastest method is SA8 (318.273 s). The slowest are KWL82a2 with 441.139 s and SGG8 with 438.909 s. All others use 323.983–430.079 s. Based on the number of black points, we see that all methods have 601 black points, except LW8 with 605 black points.

Example 5 The fifth example is a fifth degree polynomial

$$p_5(z) = z^5 - 1. \tag{71}$$

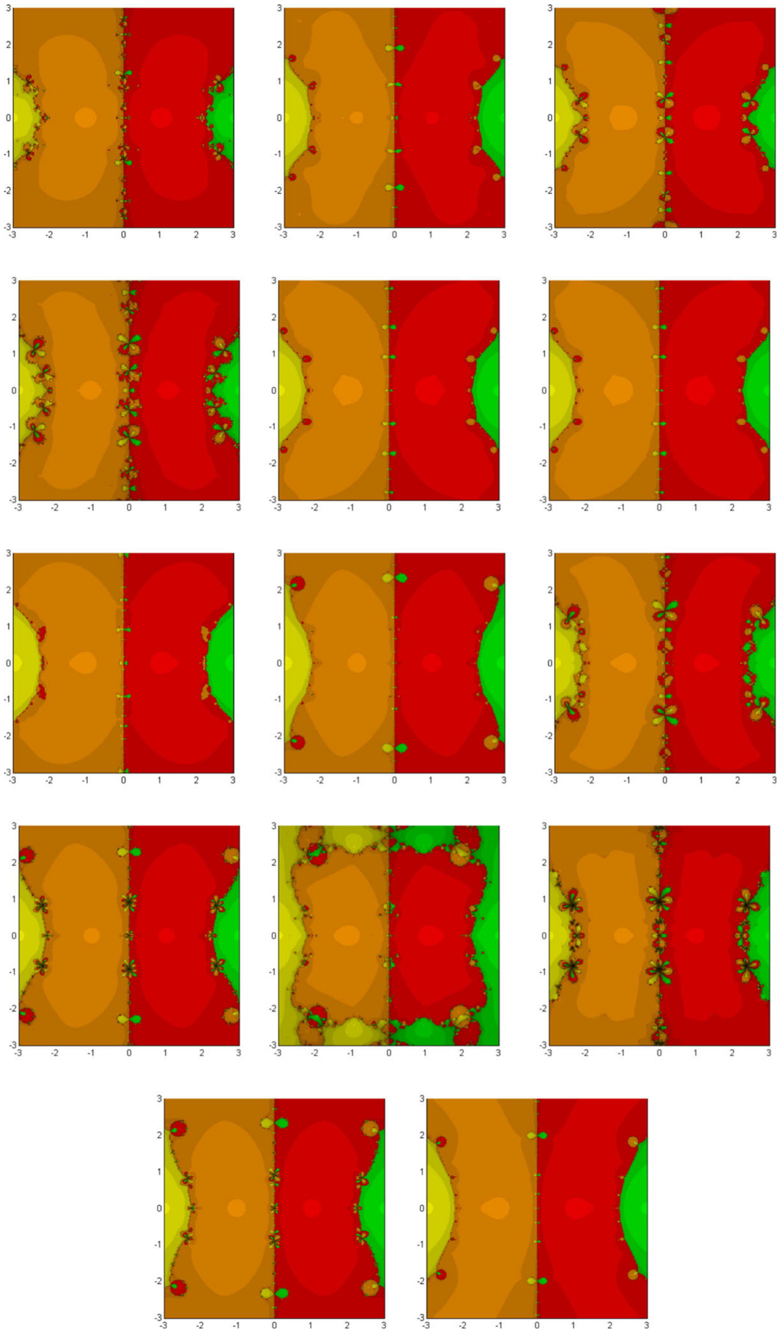

Fig. 7 The top row for NJ8 (*left*), HK8d (*center*), and HKT8 (*right*). The second row for CN8c (*left*), WL8 (*center*), and KWL81 (*right*). The third row for KWL82a2 (*left*), SA8 (*center*), and DP8 (*right*). The fourth row for CTV8 (*left*), GK with $\beta = 2$ (*center*), and LW8 (*right*). The bottom row for SAWN8 (*left*) and SGG8 (*right*) for the roots of the polynomial $z^4 - 10z^2 + 9$

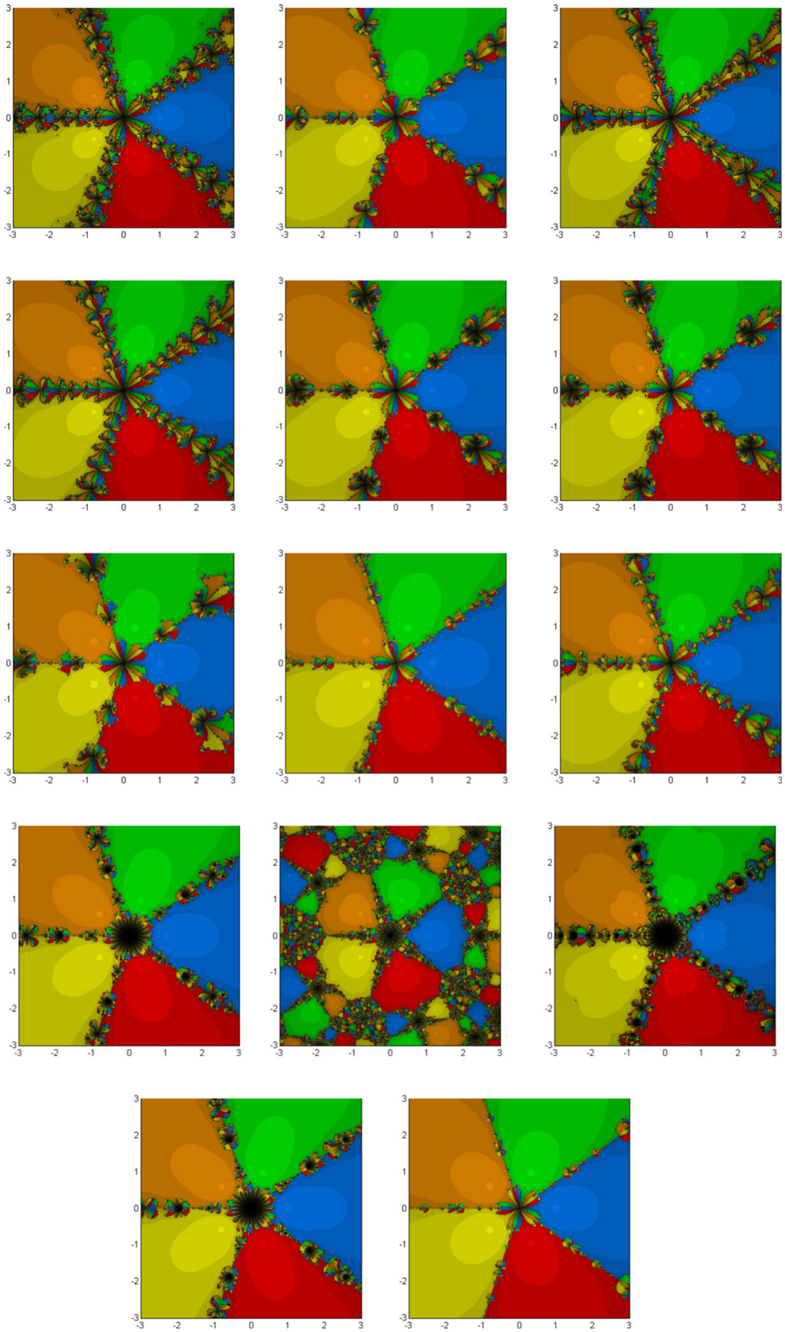

Fig. 8 The top row for NJ8 (*left*), HK8d (*center*), and HKT8 (*right*). The second row for CN8c (*left*), WL8 (*center*), and KWL81 (*right*). The third row for KWL82a2 (*left*), SA8 (*center*), and DP8 (*right*). The fourth row for CTV8 (*left*), GK with $\beta = 2$ (*center*), and LW8 (*right*). The bottom row for SAWN8 (*left*) and SGG8 (*right*) for the roots of the polynomial $z^5 - 1$

Table 4 CPU time (in seconds) required for each example (1–6) and each of the methods

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Average
NJ8	189.963	369.878	366.696	411.531	711.521	2713.263	793.81
NP8	1163.71	–	–	–	–	–	–
N8	284.624	466.022	–	–	–	–	–
N8d	249.118	391.718	398.036	–	–	–	–
HK8d	236.342	340.862	345.058	414.152	551.557	1839.49	621.24
HKT8	199.759	349.723	317.867	369.129	629.776	2207.695	678.99
KT8	286.059	493.353	–	–	–	–	–
WL8	196.858	300.145	317.977	351.002	550.995	1887.59	600.76
CN8a	278.446	435.929	435.929	–	–	–	–
CN8b	258.307	402.514	402.358	–	–	–	–
CN8c	244.267	396.492	394.979	430.079	663.363	2072.707	700.31
CN8d	267.026	445.835	419.33	–	–	–	–
KWL81	225.937	334.887	348.647	383.996	614.909	1859.485	627.977
KWL82a2	233.128	323.421	325.246	441.139	622.179	2080.149	670.877
BWR8	243.954	618.700	–	–	–	–	–
DP8	184.627	287.556	307.057	362.047	576.536	1629.165	541.165
SA8	152.381	224.969	241.178	318.273	359.240	1272.032	428.01
CTV8	162.475	254.079	254.453	323.983	437.412	1385.491	469.649
DPP8	238.229	678.355	–	–	–	–	–
GK8 ($\beta = 0$)	165.127	267.276	–	–	–	–	–
GK8 ($\beta = 2$)	161.757	269.835	268.025	387.1	678.854	–	–
GK8 ($\beta = -4/3$)	159.791	271.894	–	–	–	–	–
LW8	203.238	343.296	341.299	394.152	629.994	1935.146	641.187
SAWN8	166.172	264.219	272.799	356.478	451.935	1569.938	512.923
SGG8	237.512	332.719	372.312	438.909	509.811	1912.962	634.037
T8	282.253	743.937	–	–	–	–	–

The basins are displayed in Fig. 8. It seems that the best methods are as before HK8d, WL8, KWL81, KWL82a2, SA8, and SGG8. The data in Tables 3, 4, and 5 give a quantitative information. Based on Table 3, we find that NJ8 is the worst, requiring 25.02 function evaluations per point on average. The smallest number of function evaluations on average is for SA8 (12.11). The rest of the methods use between 12.27 and 24.37 function evaluations per point. The fastest method is SA8 (359.24 s) and the slowest is NJ8 (711.521 s). The rest use 451.935–678.854 s. In terms of black points, we find again that the worst are LW8 (3942), CTV8 (1423), and SAWN8 (1145). All other methods have between 1 and 524 black points.

Example 6 The next example is a polynomial of degree 6 with complex coefficients

$$p_6(z) = z^6 - \frac{1}{2}z^5 + \frac{11(i+1)}{4}z^4 - \frac{3i+19}{4}z^3 + \frac{5i+11}{4}z^2 - \frac{i+11}{4}z + \frac{3}{2} - 3i. \quad (72)$$

Table 5 Number of points requiring 40 iterations for each example (1–6) and each of the methods

Method	Ex1	Ex2	Ex3	Ex4	Ex5	Ex6	Average
NJ8	601	1	0	601	62	264	254.83
NP8	30825	–	–	–	–	–	–
N8	601	3	–	–	–	–	–
N8d	605	1	0	–	–	–	–
HK8d	601	1	0	601	1	0	200.67
HKT8	601	3	0	601	54	0	209.83
KT8	791	3	–	–	–	–	–
WL8	601	1	0	601	19	0	203.67
CN8a	601	1	0	–	–	–	–
CN8b	601	1	0	–	–	–	–
CN8c	601	1	0	601	16	0	203.17
CN8d	601	1	0	–	–	–	–
KWL81	601	1	0	601	17	0	203.33
KWL82a2	601	1	0	601	4	0	201.17
BWR8	601	16517	–	–	–	–	–
DP8	601	1	0	601	3	162	228
SA8	601	1	0	601	1	0	200.67
CTV8	601	14	0	601	1423	128	461.17
DPP8	601	9729	–	–	–	–	–
GK8 ($\beta = 0$)	601	7	–	–	–	–	–
GK8 ($\beta = 2$)	601	1	0	601	524	–	–
GK8 ($\beta = -4/3$)	601	2	–	–	–	–	–
LW8	601	20	0	605	3942	536	950.67
SAWN8	601	3	0	601	1145	4	392.33
SGG8	601	1	0	601	1	0	200.67
T8	617	13994	–	–	–	–	–

This is an example that was difficult for many methods. The basins are displayed in Fig. 9. The best methods seem to be KWL82a2, SA8, and SGG8. In terms of average number of function evaluations per point, SA8 is the best method with 11.57 followed by SGG8 with 11.83. The worst is NJ8 with 24.05 function evaluations per point on average. The fastest method (Table 4) is SA8 (1272.032 s) and the slowest is NJ8 (2713.263 s). LW8 and NJ8 have the highest number of black points (Table 5). It is clear that one has to use quantitative measures to distinguish between methods, since we have a different conclusion when just viewing the basins of attraction.

In order to pick the best method overall, we have averaged the results in Tables 3, 4, and 5 across the six examples. The best method based on the three criteria used is SA8. The method with the fewest number of function evaluations per point is SA8 (10.45) followed closely by KWL82a2 (10.54) and SGG8 (10.70). The fastest method is SA8 (428.01 s) followed by CTV8 (469.649 s) and SAWN8 (512.923 s).

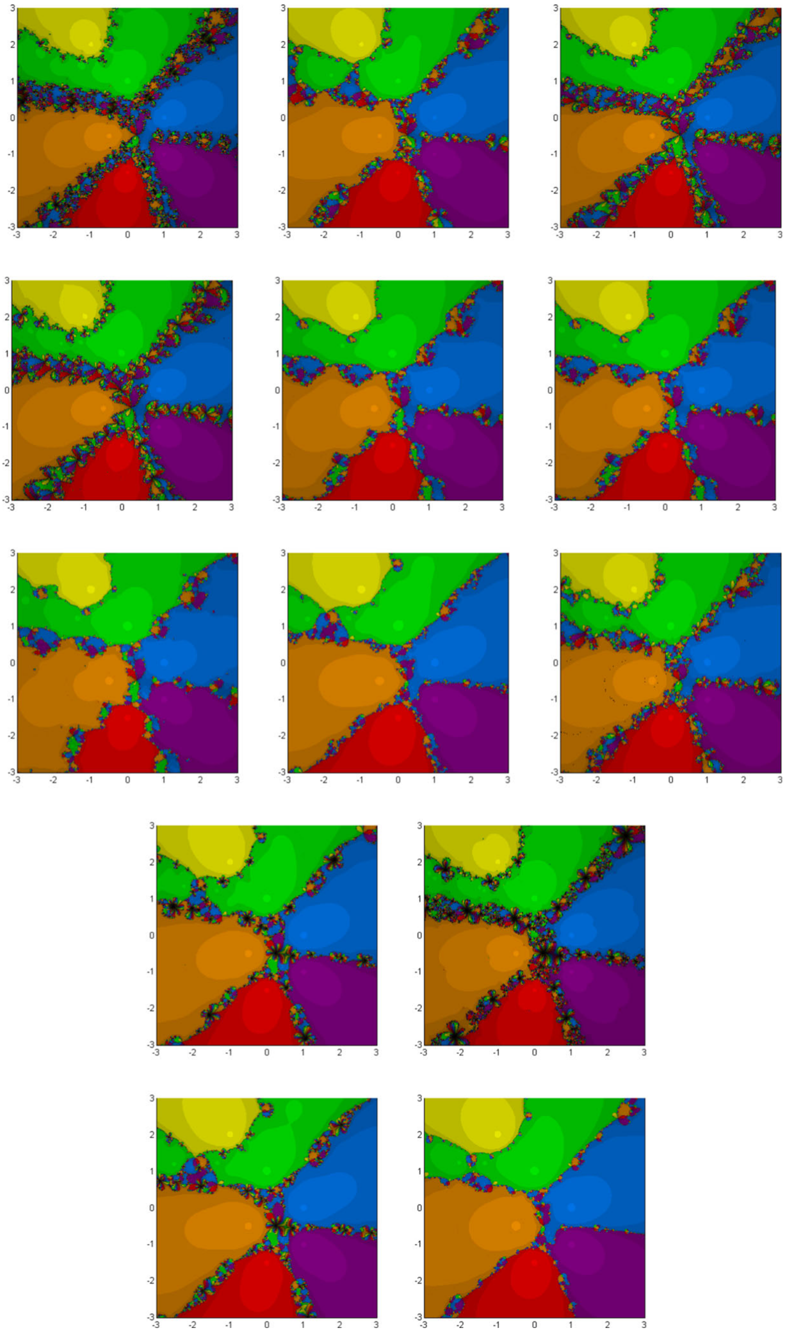

Fig. 9 The top row for NJ8 (*left*), HK8d (*center*), and HKT8 (*right*). The second row for CN8c (*left*), WL8 (*center*), and KWL81 (*right*). The third row for KWL82a2 (*left*), SA8 (*center*), and DP8 (*right*). The fourth row for CTV8 (*left*) and LW8 (*right*). The bottom row for SAWN8 (*left*) and SGG8 (*right*) for the roots of the polynomial $z^6 - \frac{1}{2}z^5 + \frac{11(i+1)}{4}z^4 - \frac{3i+19}{4}z^3 + \frac{5i+11}{4}z^2 - \frac{i+11}{4}z + \frac{3}{2} - 3i$

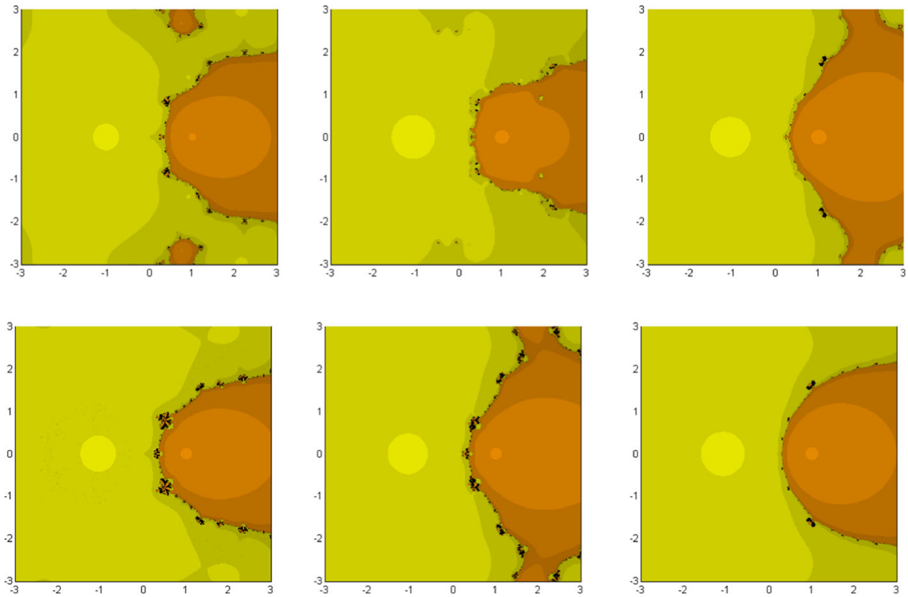

Fig. 10 The top row for HK8d (*left*), KWL82a2 (*center*), and SA8 (*right*). The second row for CTV8 (*left*), SAWN8 (*center*), and SGG8 (*right*) for the roots of the function $(e^{z+1} - 1)(z - 1)$

The methods with the least number of black points on average are HK8d, SA8, and SGG8 (200.67 points).

We now add an example with a non-polynomial function:

$$p_7(z) = (e^{z+1} - 1)(z - 1). \tag{73}$$

This example was ran on the top three performers in each category, namely HK8d, KWL82a2, SA8, CTV8, SAWN8, and SGG8. The basins are given in Fig. 10, and the results on the number of function evaluations per point on average, CPU in seconds and number of black points are summarized in Table 6.

Table 6 Results for example 7

Method	Number of function evaluation per point	CPU	Number of black points
HK8d	10.36	488.611	898
KWL82a2	9.60	486.100	439
SA8	9.12	333.967	514
CTV8	9.97	356.946	2090
SAWN8	10.21	371.891	2428
SGG8	9.36	446.100	556

4 Conclusions

We have compared the basins of several methods of order 8 using three quantitative measures and found that the best method based on all three criteria is SA8.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2005012).

References

1. Traub, J.F.: Iterative methods for the solution of equations. Prentice-Hall, Inc., Englewood Cliffs, NJ (1964)
2. Petković, M.S., Neta, B., Petković, L.D., Džunić, J.: Multipoint methods for solving nonlinear equations. Elsevier (2013)
3. Stewart, B.D.: Attractor Basins of Various Root-Finding Methods. M.S. thesis, Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA (2001)
4. Amat, S., Busquier, S., Plaza, S.: Iterative root-finding methods. Unpublished Report (2004)
5. Amat, S., Busquier, S., Plaza, S.: Review of some iterative root-finding methods from a dynamical point of view. Scientia **10**, 3–35 (2004)
6. Amat, S., Busquier, S., Plaza, S.: Dynamics of a family of third-order iterative methods that do not require using second derivatives. Appl. Math. Comput. **154**, 735–746 (2004)
7. Amat, S., Busquier, S., Plaza, S.: Dynamics of the King and Jarratt iterations. Aeq. Math. **69**, 212–223 (2005)
8. Scott, M., Neta, B., Chun, C.: Basin attractors for various methods. Appl. Math. Comput. **218**, 2584–2599 (2011)
9. Chun, C., Lee, M.Y., Neta, B., Džunić, J.: On optimal fourth-order iterative methods free from second derivative and their dynamics. Appl. Math. Comput. **218**, 6427–6438 (2012)
10. Chicharro, F., Cordero, A., Gutiérrez, J.M., Torregrosa, J.R.: Complex dynamics of derivative-free methods for nonlinear equations. Appl. Math. Comput. **219**, 7023–7035 (2013)
11. Cordero, A., García-Maimó, J., Torregrosa, J.R., Vassileva, M.P., Vindel, P.: Chaos in King's iterative family. Appl. Math. Lett. **26**, 842–848 (2013)
12. Neta, B., Scott, M., Chun, C.: Basin of attractions for several methods to find simple roots of nonlinear equations. Appl. Math. Comput. **218**, 10548–10556 (2012)
13. Neta, B., Chun, C., Scott, M.: Basins of attractions for optimal eighth order methods to find simple roots of nonlinear equations. Appl. Math. Comput. **227**, 567–592 (2014)
14. Chun, C., Neta, B., Kim, S.: On Jarratt's family of optimal fourth-order iterative methods and their dynamics. Fractals **22**, 1450013 (2014). (16 pages). doi:10.1142/S0218348X14500133
15. Chun, C., Neta, B.: On the new family of optimal eighth order methods developed by Lot et al. Numer. Algorithm. **72**, 363–376 (2016)
16. Chun, C., Neta, B.: Comparison of several families of optimal eighth order methods. Appl. Math. Comput. **274**, 762–773 (2016)
17. Argyros, I.K., Magreñan, A.A.: On the convergence of an optimal fourth-order family of methods and its dynamics. Appl. Math. Comput. **252**, 336–346 (2015)
18. Magreñan, A.A.: Different anoMalies in a Jarratt family of iterative root-finding methods. Appl. Math. Comput. **233**, 29–38 (2014)
19. Chun, C., Neta, B.: The basins of attraction of Murakami's fifth order family of methods. Applied Mathematics Comparative accepted for publication
20. Chun, C., Neta, B.: An analysis of a new family of eighth-order optimal methods. Appl. Math. Comput. **245**, 86–107 (2014)
21. Chun, C., Neta, B.: An analysis of a King-based family of optimal eighth-order methods. Am. J. Algorithm. Comput. **2**, 1–17 (2015)
22. Chun, C., Neta, B., Kozdon, J., Scott, M.: Choosing weight functions in iterative methods for simple roots. Appl. Math. Comput. **227**, 788–800 (2014)

23. Kung, H.T., Traub, J.F.: Optimal order of one-point and multipoint iterations. J. Assoc. Comput. Mach. **21**, 643–651 (1974)
24. Jarratt, P.: Some efficient fourth-order multipoint methods for solving equations. BIT **9**, 119–124 (1969)
25. Neta, B., Johnson, A.N.: High order nonlinear solver. J. Comput. Methods Sci. Eng. **8**(4-6), 245–250 (2008)
26. Neta, B., Petković, M.S.: Construction of optimal order nonlinear solvers using inverse interpolation. Appl. Math. Comput. **217**, 2448–2455 (2010)
27. Neta, B.: On a family of multipoint methods for nonlinear equations. Intern. J. Comput. Math. **9**, 353–361 (1981)
28. Wang, X., Liu, L.: Modified Ostrowski's method with eighth-order convergence and high efficiency index. Appl. Math. Lett. **23**, 549–554 (2010)
29. Ostrowski, A.M.: Solution of equations and systems of equations (1973)
30. Maheshwari, A.K.: A fourth-order iterative method for solving nonlinear equations. Appl. Math. Comput. **211**, 383–391 (2009)
31. Chun, C., Neta, B.: An analysis of a family of Maheshwari-based optimal eighth-order methods. Appl. Math. Comput. **253**, 294–307 (2015)
32. Thukral, R., Petković, M.S.: Family of three-point methods of optimal order for solving nonlinear equations. J. Comput. Appl. Math **233**, 2278–2284 (2010)
33. Bi, W., Ren, H., Wu, Q.: Three-step iterative methods with eight-order convergence for solving nonlinear equations. J. Comput. Appl. Math. **225**, 105–112 (2009)
34. Lotfi, T., Sharifi, S., Salimi, M., Siegmund, S.: A new class of three-point methods with optimal convergence order eight and its dynamics. Numer. Algor. **68**, 261–288 (2015)
35. Babajee, D.K.R., Cordero, A., Soleymani, F., Torregrosa, J.R.: On improved three-step schemes with high efficiency index and their dynamics. Numer. Algor. **65**, 153–169 (2014)
36. Cordero, A., Fardi, M., Ghasemi, M., Torregrosa, J.R.: Accelerated iterative methods for finding solutions of nonlinear equations and their dynamical behavior. Calcolo **51**, 17–30 (2014)
37. Wang, X., Liu, L.: New eighth-order iterative methods for solving nonlinear equations. J. Comput. Appl. Math. **234**, 1611–1620 (2010)
38. Sharma, J.R., Sharma, R.: A new family of modified Ostrowski's methods with accelerated eighth order convergence. Numer. Algor. **54**, 445–458 (2010)
39. Cordero, A., Torregrosa, J.R., Vassileva, M.P.: Three-step iterative methods with optimal eighth-order convergence. J. Comput. Appl. Math. **235**, 3189–3194 (2011)
40. Chun, C., Lee, M.Y.: A new optimal eighth-order family of iterative methods for the solution of nonlinear equations. Appl. Math. Comput. **223**, 506–519 (2013)
41. Bi, W., Wu, Q., Ren, H.: A new family of eight-order iterative methods for solving nonlinear equations. Appl. Math. Comput. **214**, 236–245 (2009)
42. Khan, Y., Fardi, M., Sayevand, K.: A new general eighth-order family of iterative methods for solving nonlinear equations. App. Math. Lett. **25**, 2262–2266 (2012)
43. Kou, J., Wang, X., Li, Y.: Some eighth-order root-finding three-step methods. Commun. Nonlinear Sci. Numer. Simulat. **15**, 536–544 (2010)
44. Sharma, J.R., Arora, H.: A new family of optimal eighth order methods with dynamics for nonlinear equations. Appl. Math. Comput. **273**, 924–933 (2016)
45. Cordero, A., Torregrosa, J.R., Vasileva, M.P.: A family of modified Ostrowski's methods with optimal eighth order of convergence. Appl. Math. Lett. **24**, 2082–2086 (2011)
46. Džunić, J., Petković, M.S., Petković, L.D.: A family of optimal three-point methods for solving nonlinear equations using two parametric functions. Appl. Math. Comput. **217**, 7612–7619 (2011)
47. Geum, Y.H., Kim, Y.I.: A uniparamtric family of three-step eighth-order multipoint iterative methods for simple roots. Appl. Math. Lett. **24**, 929–935 (2011)
48. Liu, L., Wang, X.: Eighth-order methods with high efficiency index for solving nonlinear equations. Appl. Math. Comput. **215**, 3449–3454 (2010)
49. Sharma, J.R., Arora, H.: An efficient family of weighted-Newton methods with optimal eighth order convergence. Appl. Math. Lett. **29**, 1–6 (2014)
50. Sharma, J.R., Guha, R.K., Gupta, P.: Improved King's methods with optimal order of convergence based on rational approximations. Appl. Math. Lett. **26**, 473–480 (2013)
51. Thukral, R.: A new eighth-order iterative method for solving nonlinear equations. Appl. Math. Comput. **217**, 222–229 (2010)