2016

Comparison of several families of optimal eighth order methods

Neta, Beny

Elsevier Inc.

Applied Mathematics and Computation 274 (2016) 762–773
http://hdl.handle.net/10945/50425

Applied Mathematics and Computation 274 (2016) 762–773

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Comparison of several families of optimal eighth order methods

Changbum Chun [a,1], Beny Neta [b,*]

[a] Department of Mathematics, Sungkyunkwan University, Suwon 16419, Republic of Korea
[b] Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA 93943, United States

ARTICLE INFO

MSC:
65H05
65B99,

Keywords:
Iterative methods
Order of convergence
Basin of attraction
Extraneous fixed points

ABSTRACT

Several families of optimal eighth order methods to find simple roots are compared to the best known eighth order method due to Wang and Liu (2010). We have tried to improve their performance by choosing the free parameters of each family using two different criteria.

Published by Elsevier Inc.

1. Introduction

There is a vast literature on the solution of nonlinear equations, see for example Ostrowski [1], Traub [2], Neta [3] and Petković et al. [4].

Lotfi et al. [5] have developed an eighth order family of optimal methods (denoted LSSS)

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

where from here on we use the following:

$$v_n = \frac{f(x_n)}{f'(x_n)}, \tag{2}$$

and

$$t_n = \frac{f(y_n)}{f(x_n)}, \tag{3}$$

$$s_n = \frac{f(z_n)}{f(x_n)}, \tag{4}$$

$$u_n = \frac{f(z_n)}{f(y_n)}. \tag{5}$$

* Corresponding author. Tel.: +1 831 656 2235; fax: +1 831 656 2355.
E-mail addresses: cbchun@skku.edu (C. Chun), bneta@nps.edu, byneta@gmail.com (B. Neta).
[1] Tel.: +82 31 299 4523; fax: +82 31 290 7033.

http://dx.doi.org/10.1016/j.amc.2015.10.092
0096-3003/Published by Elsevier Inc.

The weight functions H, K, G satisfy

$$G(0)=1, \quad G'(0)=-1, \tag{6}$$

$$K(0)=0, \quad K'(0)=2, \tag{7}$$

$$H(0)=1, \quad H'(0)=2, \quad H''(0)=10, \quad H'''(0)=72. \tag{8}$$

They also include several methods and the following families of methods in their comparative study [5]:

- Sharma2, a family of methods by Sharma and Sharma [6]

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{1-2t_n}, \\ x_{n+1} &= z_n - W(s_n)\frac{f(z_n)f[x_n,y_n]}{f[x_n,z_n]f[y_n,z_n]}, \end{aligned} \tag{9}$$

  with weight function

$$W(s_n) = 1 + \frac{s_n}{1+\alpha s_n}, \tag{10}$$

  and α is some real parameter. Sharma and Sharma [6] have used $\alpha = 1$.
- CTV, a three-parameter family of methods by Cordero et al. [7]

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{1-2t_n}, \\ x_{n+1} &= w_n - \frac{f(z_n)}{f'(x_n)}\frac{3(\beta_2+\beta_3)(w_n-z_n)}{\beta_1(w_n-z_n)+\beta_2(y_n-x_n)+\beta_3(z_n-x_n)}, \end{aligned} \tag{11}$$

  where

$$w_n = z_n - \frac{f(z_n)}{f'(x_n)}\left(\frac{1-t_n}{1-2t_n} + \frac{1}{2}\frac{u_n}{1-2u_n}\right)^2, \tag{12}$$

  and β_1, β_2, and β_3 are real parameters with $\beta_2 + \beta_3 \neq 0$.
  Remark: Cordero et al. [7] have used $\beta_1 = \beta_3 = 0$ and $\beta_2 = 1$.
- CL, a two-parameter family of methods by Chun and Lee [8]

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{(1-t_n)^2}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)}\frac{1}{(1-H(t_n)-J(s_n)-P(u_n))^2}, \end{aligned} \tag{13}$$

  where the weight functions should satisfy the following conditions to guarantee eighth order:

$$H(0)=0, \quad H'(0)=1, \quad H''(0)=1, \quad H'''(0)=-3, \tag{14}$$

$$J(0)=0,\ J'(0)=\frac{1}{2}, \quad P(0)=0, \quad P'(0)=\frac{1}{2}. \tag{15}$$

  Remark: Chun and Lee [8] have used the following weight functions

$$\begin{aligned} H(t_n) &= -\beta-\gamma+t_n+t_n^2/2-t_n^3/2, \\ J(s_n) &= \beta + s_n/2, \\ P(u_n) &= \gamma + u_n/2, \end{aligned} \tag{16}$$

  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:

$$\begin{aligned} J(t) &= \frac{a_1+b_1 t}{1+\delta_1 t}, \\ P(t) &= \frac{a_2+b_2 t}{1+\delta_2 t}, \end{aligned}$$

$$H(t) = \frac{a_3 + b_3 t + c_3 t^2}{1 + \delta_3 t + g_3 t^2}. \tag{17}$$

These functions satisfying the conditions (14) and (15) are given by

$$J(t) = \frac{1}{2}\frac{t}{1 + \delta_1 t}, \tag{18}$$

$$P(t) = \frac{1}{2}\frac{t}{1 + \delta_2 t}, \tag{19}$$

$$H(t) = \frac{1}{2}\frac{2t + (3 - 4g_3)t^2}{1 + (1 - 2g_3)t + g_3 t^2}. \tag{20}$$

- BRW, a family of methods by Bi et al. [9]

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1 - t_n/2}{1 - 5t_n/2}, \\ x_{n+1} &= z_n - H(s_n)\frac{f(z_n)}{f[z_n, y_n] + f[z_n, x_n, x_n](z_n - y_n)}, \end{aligned} \tag{21}$$

where the weight function should satisfy the following condition to guarantee eighth order:

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

Remark: Bi et al. [9] have used

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

with α, a non-zero real number, chosen as unity.

We consider here the more general weight function

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

This function satisfying the condition (22) is given by

$$H(t) = \frac{1 + (c + 2)t}{1 + ct + gt^2}. \tag{25}$$

- TP, a two-parameter family of methods by Thukral and Petković [10]

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1 + \beta t_n}{1 + (\beta - 2)t_n}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{f'(x_n)}\left(H(t_n) + \frac{u_n}{1 - \alpha u_n} + 4s_n\right), \end{aligned} \tag{26}$$

with weight function

$$H(t_n) = \frac{5 - 2\beta - (2 - 8\beta + 2\beta^2)t_n + (1 + 4\beta)t_n^2}{5 - 2\beta - (12 - 12\beta + 2\beta^2)t_n}, \tag{27}$$

for some real parameters α and β.
Remark: Thukral and Petković [10] have used this method with $\alpha = \beta = 0$, but the method is more general and includes 3 weight functions, $H(t_n)$, $\psi(u_n)$, and $W(s_n)$.

In [11] we have compared several members of the family LSSS to the methods cited there and to the method by Wang and Liu [12] denoted WL which showed good results in our previous work (see, for example, Chun and Neta [11,13]. The method WL is as follows:

$$\begin{aligned} y_n &= x_n - v_n, \\ z_n &= y_n - \frac{f(y_n)}{f'(x_n)}\frac{1}{1 - 2t_n}, \\ x_{n+1} &= z_n - \frac{f(z_n)}{2(f[x_n, z_n] - f[x_n, y_n]) + f[y_n, z_n] + \frac{y_n - z_n}{y_n - x_n}(f[x_n, y_n] - f'(x_n))}. \end{aligned} \tag{28}$$

In the next section we discuss two criteria to choose the parameters in the families Sharma2, CTV, CL, BRW and TP. These criteria were developed in [13].

Table 1
The function H_f for each of the 5 families of methods.

Methods	H_f
Sharma2	$1+\frac{t_n}{1-2t_n}+W(s_n)\frac{s_n f[x_n,y_n]f'(x_n)}{f[x_n,z_n]f[y_n,z_n]}$
CTV	$1+\frac{t_n}{1-2t_n}+s_n\left(\frac{1-t_n}{1-2t_n}+\frac{1}{2}\frac{u_n}{1-2u_n}\right)^2+s_n\frac{3(\beta_2+\beta_3)(w_n-z_n)}{\beta_1(w_n-z_n)+\beta_2(y_n-x_n)+\beta_3(z_n-x_n)}$
CL	$1+\frac{t_n}{(1-t_n)^2}+\frac{s_n}{(1-H(t_n)-J(s_n)-P(u_n))^2}$
BRW	$1+\frac{t_n(1-t_n/2)}{1-5t_n/2}+\frac{s_n H(s_n)f'(x_n)}{f[z_n,y_n]+f[z_n,x_n,x_n](z_n-y_n)}$
TP	$1+\frac{t_n(1+\beta t_n)}{1+(\beta-2)t_n}+s_n\left(H(t_n)+\frac{u_n}{1-\alpha u_n}+4s_n\right)$

2. Extraneous fixed points

In solving a nonlinear equation iteratively we are looking for fixed points which are zeros of the given nonlinear function. Many multipoint iterative methods have fixed points that are not zeros of the function of interest. Thus, it is imperative to investigate the number of extraneous fixed points, their location and their properties.

The parameters can be chosen to position the extraneous fixed points on the imaginary axis or, at least, close to that axis, (see, for example, Chun et al. [14] and Chun and Neta [15]).

In order to find the extraneous fixed points, we rewrite the families of methods of interest in the form

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}H_f(x_n,y_n,z_n), \tag{29}$$

where the function H_f for each family of methods is given in Table 1.

We have searched the parameter space and found that the extraneous fixed points are not on the imaginary axis. We have considered one measure of closeness to the imaginary axis, denoted by d, and another measure of averaged stability of the extraneous fixed points, denoted by A. These measures are defined below. We have experimented with those members from the parameter space.

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

$$d=\max_{z_i\in E}|Re(z_i)|. \tag{30}$$

We look for the parameters which attain the minimum of the function d given in (30).

Another method to choose the parameters is by considering the stability of $z\in E$ defined by

$$dq(z)=\frac{dq}{dz}(z), \tag{31}$$

where q is the iteration function of (29). We define a function called the averaged stability value of the set E by

$$A=\frac{\sum\limits_{z_i\in E}|dq(z_i)|}{n}. \tag{32}$$

The smaller A becomes, the less chaotic the basin of attraction tends to.

In Table 2 we list the values of the parameters that minimize d or A for each method of the following Sharma2, CTV, CL, BRW and TP. We have denoted the method with a suffix of d or A, respectively. We also included in the table the methods with the parameters suggested by their authors.

3. Numerical experiments

The Basin of Attraction is a method to visually understand how a method behaves as a function of the various starting points. This idea was started by Stewart [16] and continued in the work of Amat et al. [17–20], Scott et al. [21], Chun et al. [22], Chun and Neta [23], Chicharro et al. [24], Cordero et al. [25], Neta et al. [26,27], Argyros and Magreñan [28], Magreñan [29], Geum et al. [30] and Chun et al. [31]. The only papers comparing basins of attraction for methods to obtain multiple roots are due to Neta et al. [32], Neta and Chun [33,34], Chun and Neta [35–37].

We have used the 5 families of methods Sharma2, CTV, CL, BRW, TP and WL for 5 different polynomials. The choice of the parameters in the families used is based on the analysis in the previous section. 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

Table 2
Value of the parameters used originally and the ones that minimize either d or A for each of the 5 families of methods.

Methods	Parameter 1	Parameter 2	Parameter 3
Sharma2	$\alpha = 1$	-	-
Sharma2A	$\alpha = 1.4$	-	-
Sharma2d	$\alpha = -2.1$	-	-
CTV	$\beta_1 = 0$	$\beta_2 = 1$	$\beta_3 = 0$
CTVA	$\beta_1 = 1.6$	$\beta_2 = -1.2$	$\beta_3 = 1.9$
CTVd	$\beta_1 = 3.3$	$\beta_2 = -3.2$	$\beta_3 = -1.0$
CL	$\beta = 0$	$\gamma = 0$	-
CLA	$\delta_1 = -0.2$	$\delta_2 = 0.5$	$g_3 = 0.9$
CLd	$\delta_1 = -0.3$	$\delta_2 = -1.2$	$g_3 = 2.3$
BRW	$\alpha = 1$	-	-
BRWA	$c = 2.3$	$g = 0.6$	-
BRWd	$c = 0.7$	$g = -0.3$	-
TP	$\beta = 0$	$\alpha = 0$	-
TPA	$\beta = -1.6$	$\alpha = -1.7$	-
TPd	$\beta = -1$	$\alpha = 0.9$	-

Table 3
Average number of iterations per point for each example (1–5) and each of the 16 methods.

Methods	Ex1	Ex2	Ex3	Ex4	Ex5	Average
Sharma2	2.45	2.97	3.85	4.08	4.19	3.51
Sharma2A	2.45	2.99	3.75	3.78	3.62	3.32
Sharma2d	2.44	2.94	3.72	3.74	3.56	3.28
CTV	2.43	3.44	5.17	5.83	5.07	4.39
CTVA	2.51	3.39	4.85	5.24	4.69	4.14
CTVd	2.37	3.27	4.42	4.91	4.47	3.89
CL	2.54	3.36	4.23	4.38	4.16	3.74
CLA	2.62	3.79	5.71	6.44	5.73	4.86
CLd	2.35	3.71	7.55	9.09	7.78	6.09
BRW	2.33	4.88	4.77	5.20	4.48	4.33
BRWA	2.30	4.12	4.71	5.31	4.43	4.18
BRWd	2.35	4.34	4.95	5.71	4.62	4.42
TP	2.75	3.98	6.41	6.47	5.70	5.06
TPA	5.17	32.81	*	*	*	*
TPd	2.65	14.48	*	*	*	*
WL	2.27	2.71	3.52	4.04	3.44	3.20

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 iterations required per point and the number of points requiring 40 iterations.

Example 1. In our first example, we have taken the polynomial

$$p_1(z) = z^2 - 1 \tag{33}$$

whose roots are $z = \pm 1$. In Fig. 1 we have presented the basins for the 16 methods. It is clear from the figure that WL is the best method (bottom subplot). The parameters based on the d criterion for closeness has improved the basins (third column versus first) in most cases. The parameters based on the A criterion made the situation worse for TP. In order to get a quantitative comparison, we have collected the average number of iterations per point in Table 3, the CPU time for each method and each example in Table 4 and the number of points requiring 40 iterations in Table 5. It is clear that WL requires the least number (2.27). All other methods (except TPA) require between 2.30 and 2.75 iterations per point on average. In terms of CPU time WL is the fastest with 184.82s. All other methods use more than 400 s. The slowest is TPA with 1350.17 s. In terms of the number of points requiring 40 iterations (see Table 5) we find that TPA is the worst with 721 points. All other methods have 601–605 points requiring 40 iterations.

Example 2. Our next example is a cubic polynomial having the three roots of unity,

$$p_2(z) = z^3 - 1. \tag{34}$$

The basins of attraction are plotted in Fig. 2. Again the worst are TPA and TPd and the best are WL and the 3 versions of Sharma2. Therefore we will not show TPA and TPd in the rest of the examples. The average number of iterations per point is the lowest for WL (2.71) followed by Sharma2d. The worst is TPA (32.81) and TPd (14.48). The fastest method is WL (294.54 s) and the slowest is TPA (13044.70 s). The methods with the fewest number of points requiring 40 iterations are: WL, Sharma2A, Sharma2d, and CL. The worst are: TPA (290,803 points), TPd (46,413 points), and BRW (19,882 points).

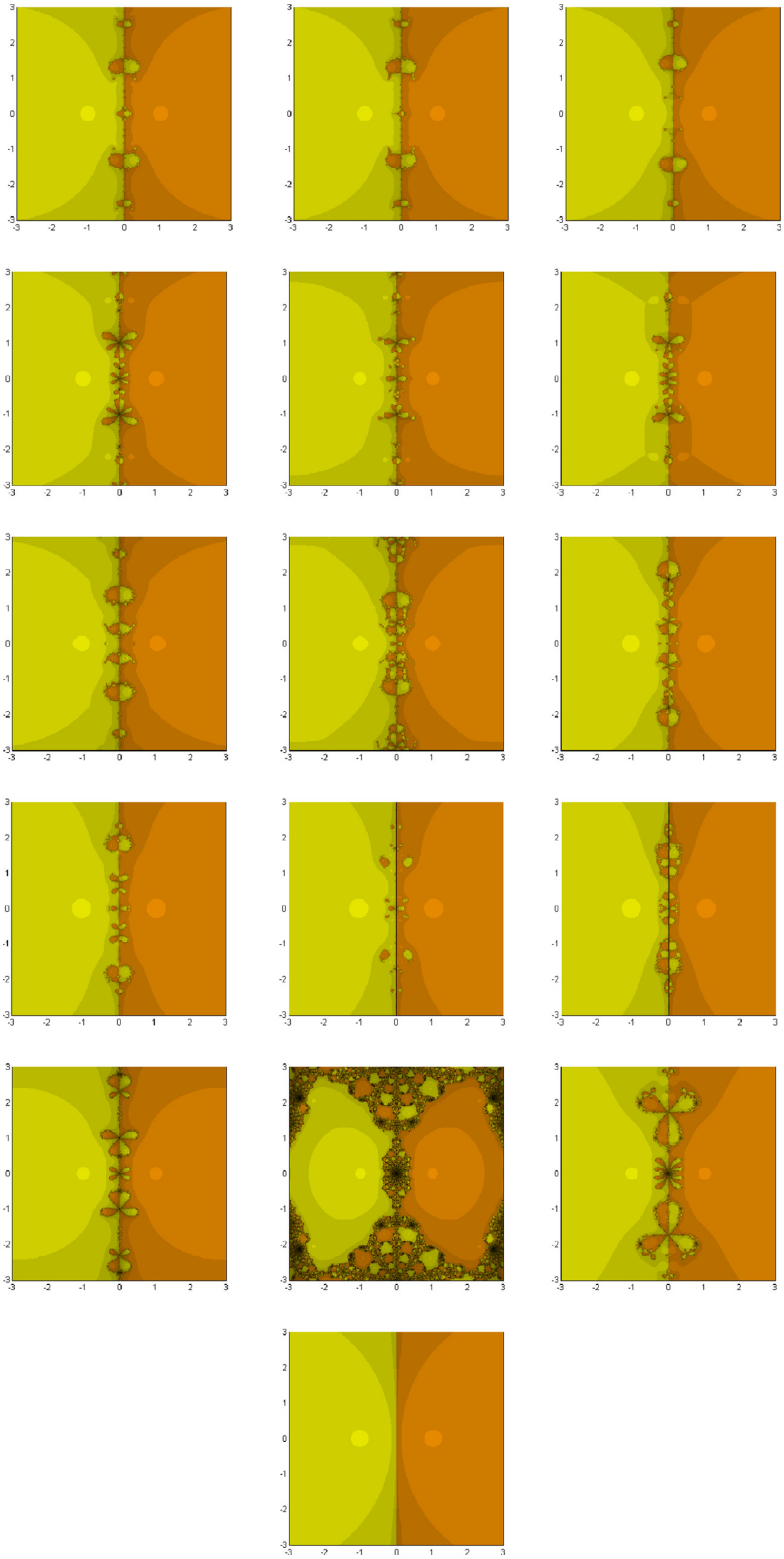

Fig. 1. The top row for Sharma2, second row for CTV, third row for CL, fourth row for BRW, fifth row for TP and on the last row WL for the roots of the polynomial $z^2 - 1$. The left column for the family with the original parameters, center column with the parameters based on A and on the right column with the parameters based on d.

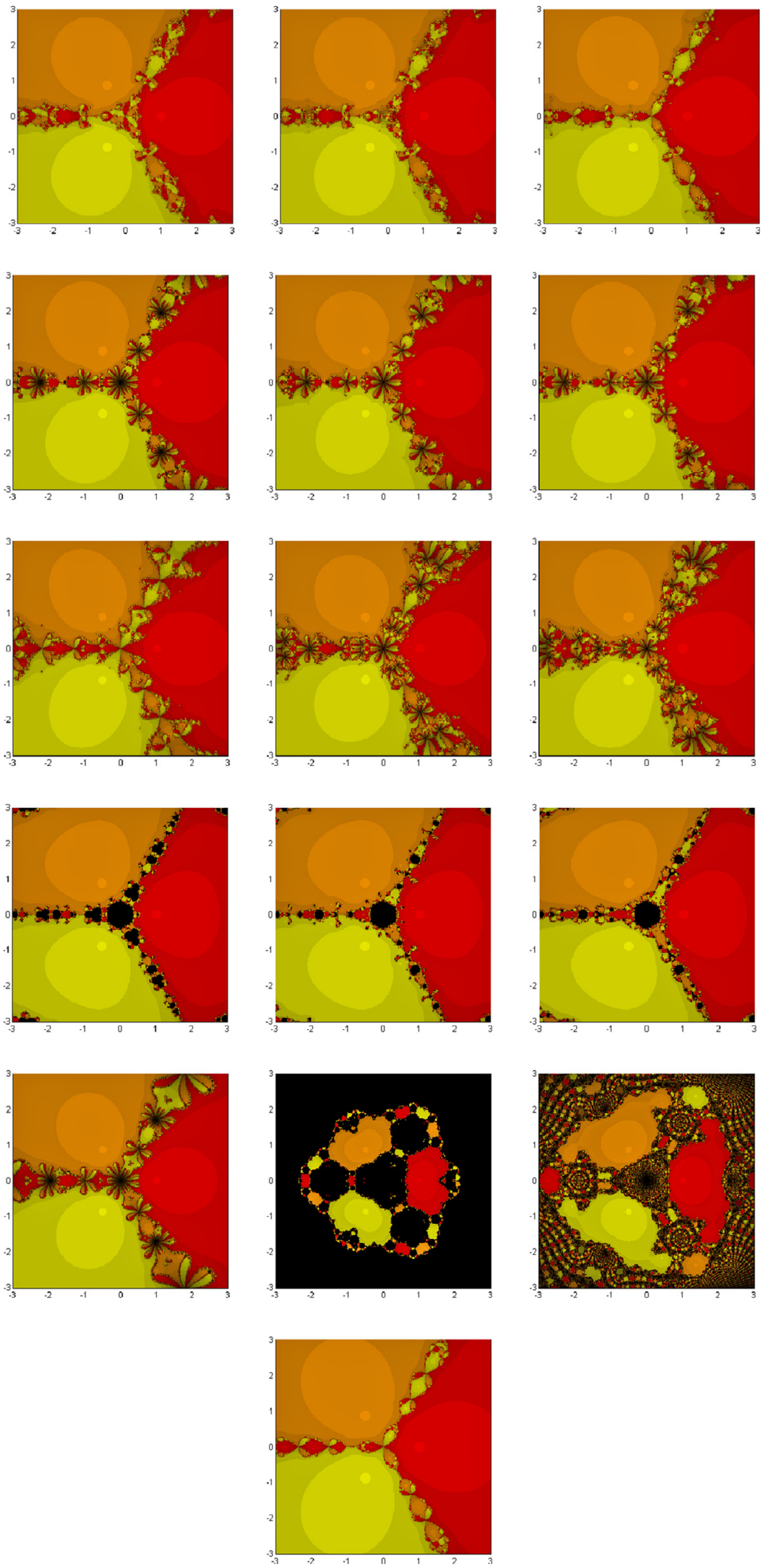

Fig. 2. The top row for Sharma2, second row for CTV, third row for CL, fourth row for BRW, fifth row for TP and on the last row WL for the roots of the polynomial $z^3 - 1$. The left column for the family with the original parameters, center column with the parameters based on A and on the right column with the parameters based on d.

Table 4
CPU time (in seconds) required for each example (1–5) and each of the 16 methods.

Methods	Ex1	Ex2	Ex3	Ex4	Ex5	Average
Sharma2	514.87	767.76	1179.62	1325.55	4440.29	1645.62
Sharma2A	485.17	778.45	1032.53	1244.25	3620.21	1432.12
Sharma2d	481.09	765.78	1026.00	1213.59	3837.89	1464.87
CTV	526.25	939.61	1677.62	2133.25	5275.47	2110.44
CTVA	514.37	945.01	1561.34	1787.19	5006.96	1962.98
CTVd	491.59	910.67	1435.41	1697.23	4885.11	1884
CL	466.25	770.94	1059.69	1274.31	3562.67	1426.77
CLA	545.41	993.75	1595.37	2213.58	4856.26	2040.87
CLd	490.92	920.50	2014.62	1275.90	3872.62	1714.91
BRW	483.64	1246.27	1370.36	1739.12	2667.90	1501.46
BRWA	489.73	1113.53	1364.59	1821.73	5928.92	2143.70
BRWd	504.91	1233.19	1443.52	1960.22	5580.55	2144.48
TP	566.11	1062.47	2561.19	2004.27	5355.86	2309.98
TPA	1350.17	13044.70	*	*	*	*
TPd	455.97	3419.91	*	*	*	*
WL	184.82	294.54	404.83	599.11	1888.92	674.45

Table 5
Number of points requiring 40 iterations for each example (1–5) and each of the 16 methods.

Methods	Ex1	Ex2	Ex3	Ex4	Ex5	Average
Sharma2	601	55	1201	17	4705	1315.8
Sharma2A	601	1	1201	6	166	395
Sharma2d	601	1	1201	1	157	392.2
CTV	605	69	4765	9846	2308	3518.6
CTVA	601	27	2037	4056	552	1454.6
CTVd	601	5	1609	3316	431	1192.4
CL	601	1	1201	5	0	361.6
CLA	601	19	2025	4322	3301	2053.6
CLd	601	57	4521	19442	11496	7223.4
BRW	601	19882	2381	4364	195	5484.6
BRWA	601	12509	2913	5837	266	4425.2
BRWd	601	14235	3717	8416	336	5461
TP	601	159	7109	11156	2666	4338.2
TPA	721	290803	*	*	*	*
TPd	601	46413	*	*	*	*
WL	601	1	1201	19	0	364.4

Example 3. Our next example is a quartic polynomial having the four roots of unity,

$$p_3(z) = z^4 - 1, \tag{35}$$

where the roots are symmetrically located on the axes. In some sense this is similar to the first example, since in both cases we have an even number of roots. The basins of attraction are given in Fig. 3. Note that now we have WL on the right of TP on the last row, since we have decided not to run TPA and TPd. The best is WL followed by the 3 versions of Sharma2, and CL. The worst is CLd. We can now consult the tables to see that WL and the 3 versions of Sharma2 require less than 4 iterations per point on average while CLd requires the most with 7.55 iterations per point on average. WL is again the fastest (see Table 4) with 404.83s. All other methods use over 1026 s with TP being the slowest (2561.19s). Similar conclusion from Table 5, namely WL, the 3 versions of Sharma2 and CL have 1201 points requiring 40 iterations and TP has the most points (7109).

Example 4. The fourth example is a polynomial

$$p_4(z) = z^5 - 1. \tag{36}$$

The plots of the basins are given in Fig. 4. The best are Sharma2d and Sharma2A followed by WL. In terms of the average number of iterations per point (see Table 3) we have the same conclusion. CLd and TP require the most number of iterations per point on average. WL is the fastest (see Table 4) followed by Sharma2d, Sharma2A and CL. The slowest are CLA (2213.58 s) and CTV (2133.25 s). In terms of the number of points requiring 40 iterations, the best are Sharma2d, CL, Sharma2A, Sharma2 and WL, the worst is TP (11,156 points).

Example 5. Our last example is a polynomial with complex coefficients

$$p_5(z) = z^6 - \frac{1}{2}z^5 + \frac{11}{4}(1+i)z^4 - \frac{1}{4}(19+3i)z^3 + \frac{1}{4}(11+5i)z^2 - \frac{1}{4}(11+i)z + \frac{3}{2} - 3i. \tag{37}$$

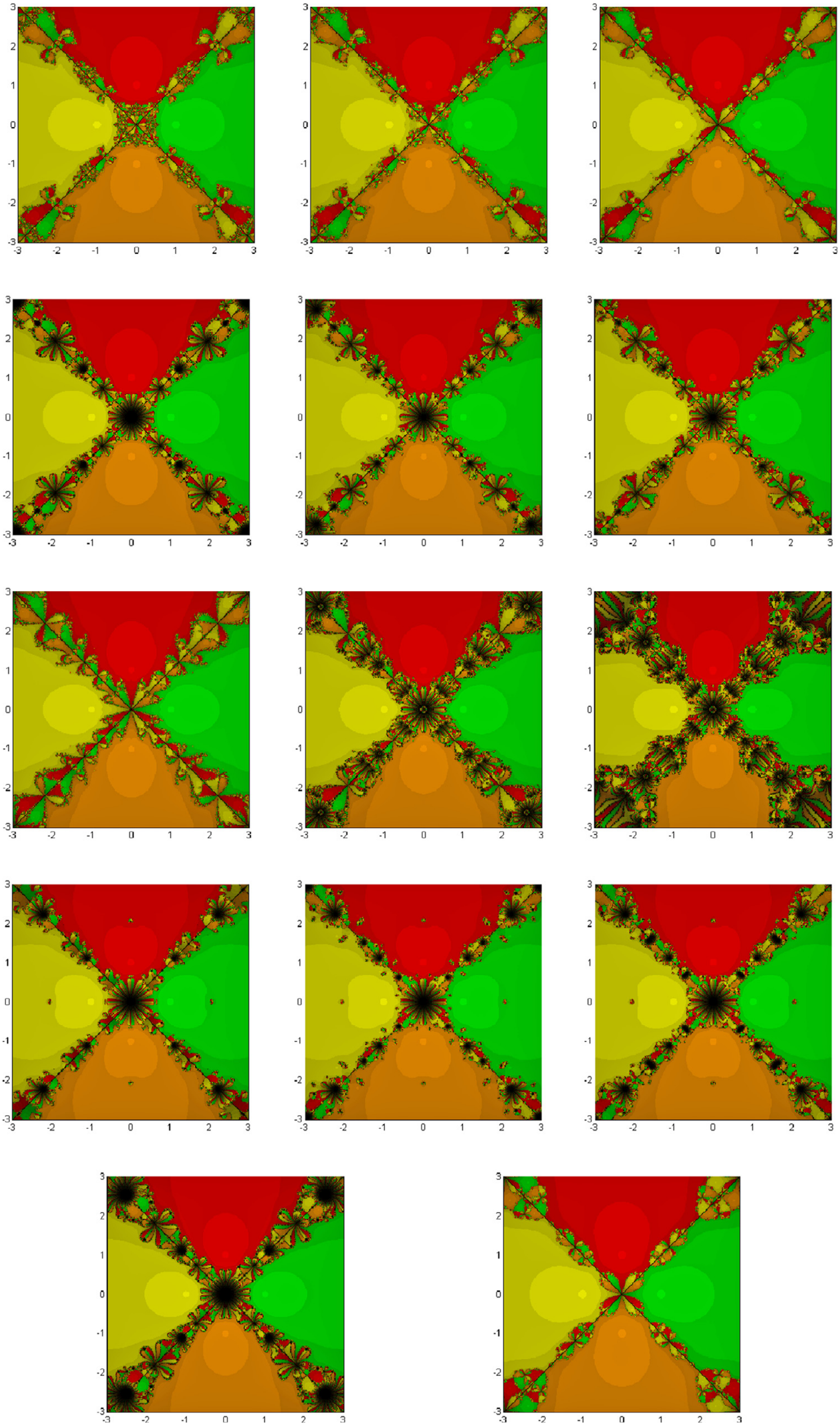

Fig. 3. The top row for Sharma2, second row for CTV, third row for CL, fourth row for BRW, fifth row for TP on the left and WL on the right for the roots of the polynomial $z^4 - 1$. The left column for the family with the original parameters, center column with the parameters based on A and on the right column with the parameters based on d.

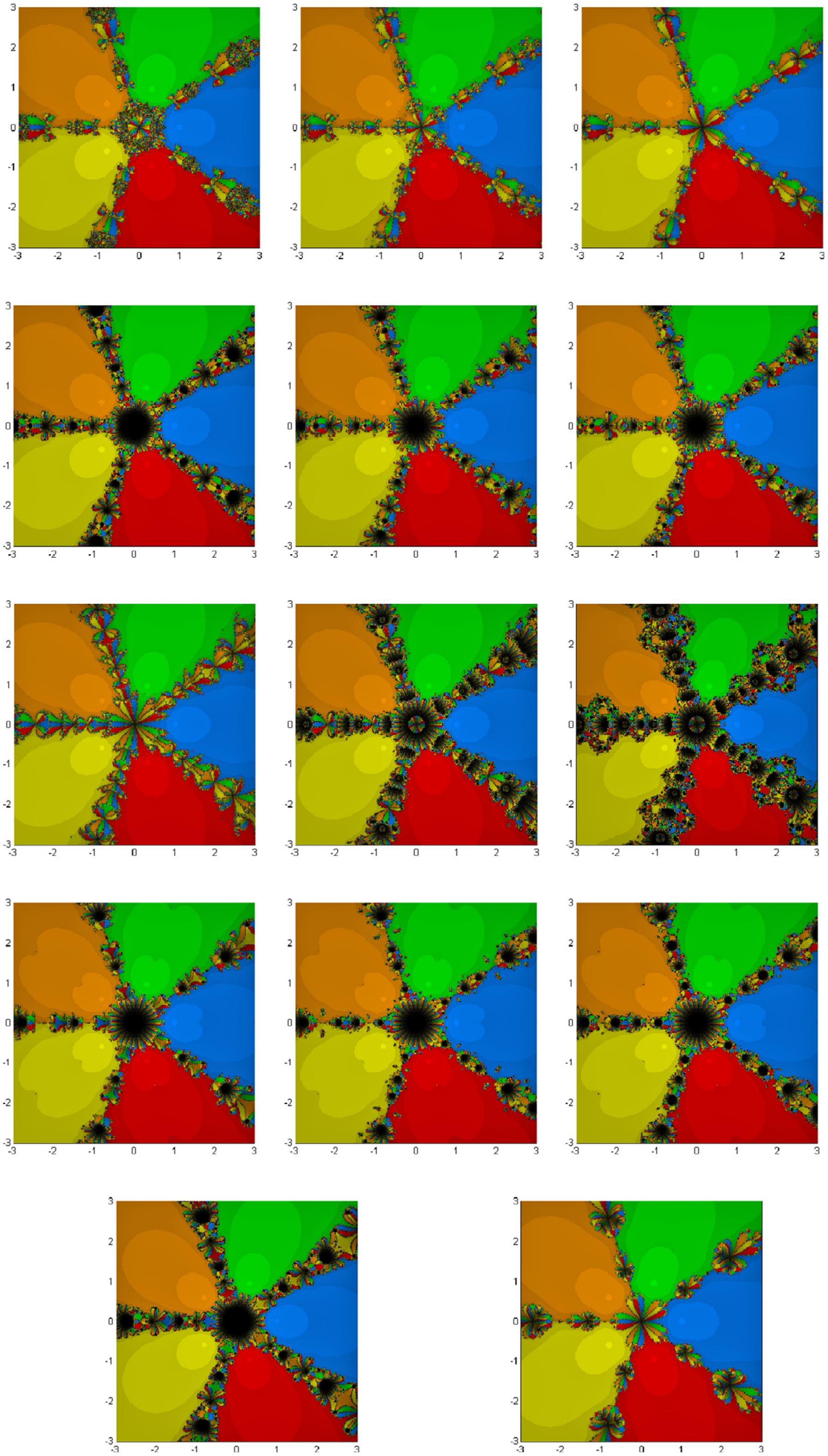

Fig. 4. The top row for Sharma2, second row for CTV, third row for CL, fourth row for BRW, fifth row for TP on the left and WL on the right for the roots of the polynomial $z^5 - 1$. The left column for the family with the original parameters, center column with the parameters based on A and on the right column with the parameters based on d.

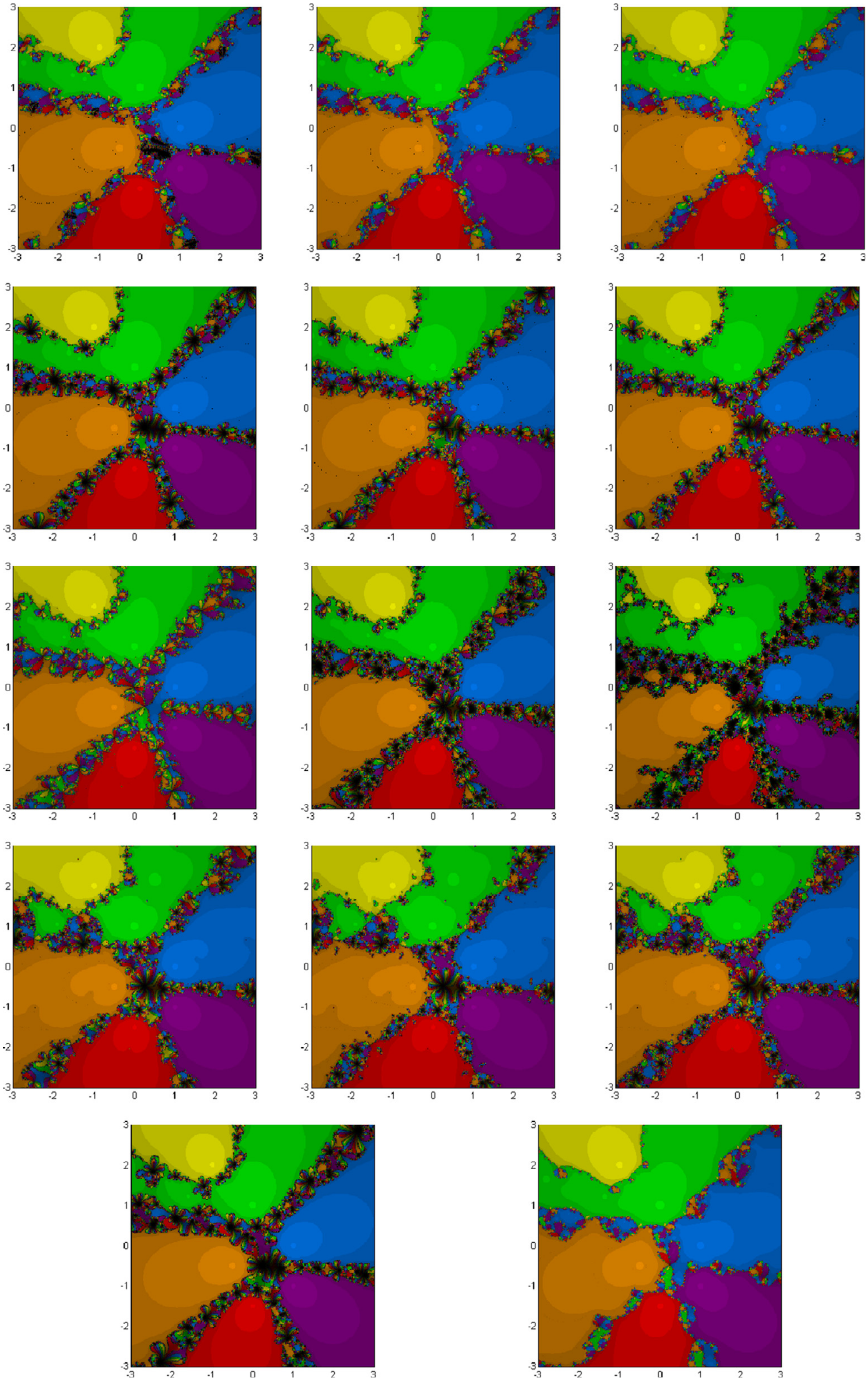

Fig. 5. The top row for Sharma2, second row for CTV, third row for CL, fourth row for BRW, fifth row for TP on the left and WL on the right for the roots of the polynomial with complex coefficients $z^6 - \frac{1}{2}z^5 + \frac{11}{4}(1+i)z^4 - \frac{1}{4}(19+3i)z^3 + \frac{1}{4}(11+5i)z^2 - \frac{1}{4}(11+i)z + \frac{3}{2} - 3i$. The left column for the family with the original parameters, center column with the parameters based on A and on the right column with the parameters based on d.

This example was the hardest for many iterative methods as we found out in our previous work. WL, Sharma2A and Sharma2d are best, as can be seen from Fig. 5. From Table 3 we find that WL, Sharma2d and Sharma2A require 3.44–3.62 iterations per point on average. The worst are CLd, CLA and TP with 5.70–7.78 iterations per point on average. WL is the fastest with 1888.92s, followed by BRW. All other methods use over 3500seconds with BRWA the slowest (5928.92 s).

Conclusions

In Tables 3–5 we averaged the results across all 5 examples. Based on Table 3, we find that WL, Sharma2d and Sharma2A are best (3.2–3.32 iterations per point on average). The others require 3.51–5.06 iterations per point on average. The fastest method on all examples is WL (674.45 s) followed by CL (1426.77 s), Sharma2A (1432.12 s) and Sharma2d (1464.87 s). Similar conclusions one can find in Table 5. The lowest number of points requiring 40 iterations is for CL, WL, Sharma2d and Sharma2A in that order. All the others have at least 1192 such points on average. Overall we can say that WL is best and followed by the 3 versions of Sharma2 and CL in places 2–5 (depending on the criterion used).

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). The first author thanks the Applied Mathematics Department at the Naval Postgraduate School for hosting him during the years.

References

[1] A.M. Ostrowski, Solution of Equations in Euclidean and Banach Space, Academic Press, New York, 1973.
[2] J.F. Traub, Iterative Methods for the Solution of Equations, Chelsea Publishing Company, New York, 1977.
[3] B. Neta, Numerical Methods for the Solution of Equations, Net-A-Sof, California, 1983.
[4] M.S. Petković, B. Neta, L.D. Petković, J. Džunić, Multipoint Methods for Solving Nonlinear Equations, Elsevier, Waltham, MA, 2013.
[5] T. Lotfi, S. Sharifi, M. Salimi, S. Siegmund, A new class of three-point methods with optimal convergence order eight and its dynamics, Numer. Algorithms 68 (2015) 261–288.
[6] J.R. Sharma, R. Sharma, A new family of modified Ostrowski's methods with accelerated eighth order convergence, Numer. Algorithms 54 (2010) 445–458.
[7] A. Cordero, J.R. Torregrosa, M.P. Vassileva, Three-step iterative methods with optimal eighth-order convergence, J. Comput. Appl. Math. 235 (2011) 3189–3194.
[8] C. Chun, M.Y. Lee, A new optimal eighth-order family of iterative methods for the solution of nonlinear equations, Appl. Math. Comput. 223 (2013) 506–519.
[9] W. Bi, H. Ren, Q. Wu, Three-step iterative methods with eighth-order convergence for solving nonlinear equations, J. Comput. Appl. Math. 225 (2009) 105–112.
[10] R. Thukral, M.S. Petković, A family of three-point methods of optimal order for solving nonlinear equations, J. Comput. Appl. Math. 233 (2010) 2278–2284.
[11] C. Chun, B. Neta, On the new family of optimal eighth order methods developed by Lotfi et al., Numer. Algorithms, accepted for publication.
[12] X. Wang, L. Liu, Modified Ostrowski's method with eighth-order convergence and high efficiency index, Appl. Math. Lett. 23 (2010) 549–554.
[13] C. Chun, B. Neta, An analysis of a King-based family of optimal eighth-order methods, Am. J. Algorithms Comput. 2 (2015) 1–17.
[14] C. Chun, B. Neta, J. Kozdon, M. Scott, Choosing weight functions in iterative methods for simple roots, Appl. Math. Comput. 227 (2014) 788–800.
[15] C. Chun, B. Neta, An analysis of a family of Maheshwari-based optimal eighth order methods, Appl. Math. Comput. 253 (2015) 294–307.
[16] B.D. Stewart, Attractor basins of various root-finding methods, Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA, June 2001 (M.s. thesis).
[17] S. Amat, S. Busquier, S. Plaza, Iterative Root-Finding Methods, unpublished report (2004).
[18] S. Amat, S. Busquier, S. Plaza, Review of some iterative root-finding methods from a dynamical point of view, Scientia 10 (2004) 3–35.
[19] S. Amat, S. Busquier, S. Plaza, Dynamics of a family of third-order iterative methods that do not require using second derivatives, Appl. Math. Comput. 154 (2004) 735–746.
[20] S. Amat, S. Busquier, S. Plaza, Dynamics of the King and Jarratt iterations, Aequ. Math. 69 (2005) 212–223.
[21] M. Scott, B. Neta, C. Chun, Basin attractors for various methods, Appl. Math. Comput. 218 (2011) 2584–2599.
[22] C. Chun, M.Y. Lee, B. Neta, J. Džunić, On optimal fourth-order iterative methods free from second derivative and their dynamics, Appl. Math. Comput. 218 (2012) 6427–6438.
[23] C. Chun, B. Neta, An analysis of a new family of eighth-order optimal methods, Appl. Math. Comput. 245 (2014) 86–107.
[24] F. Chicharro, A. Cordero, J.M. Gutiérrez, J.R. Torregrosa, Complex dynamics of derivative-free methods for nonlinear equations, Appl. Math. Comput. 219 (2013) 7023–7035.
[25] A. Cordero, J. García-Maimó, J.R. Torregrosa, M.P. Vassileva, P. Vindel, Chaos in King's iterative family, Appl. Math. Lett. 26 (2013) 842–848.
[26] B. Neta, M. Scott, C. Chun, Basin of attractions for several methods to find simple roots of nonlinear equations, Appl. Math. Comput. 218 (2012) 10548–10556.
[27] B. Neta, C. Chun, M. Scott, Basins of attraction for optimal eighth order methods to find simple roots of nonlinear equations, Appl. Math. Comput. 227 (2014) 567–592.
[28] I.K. Argyros, A.A. Magreñan, On the convergence of an optimal fourth-order family of methods and its dynamics, Appl. Math. Comput. 252 (2015) 336–346.
[29] A.A. Magreñan, Different anomalies in a Jarratt's family of iterative root-finding methods, Appl. Math. Comput. 233 (2014) 29–38.
[30] Y.H. Geum, Y.I. Kim, B. Neta, On developing a higher-order family of double-Newton methods with a bivariate weighting function, Appl. Math. Comput. 254 (2015) 277–290.
[31] C. Chun, B. Neta, S. Kim, On Jarratt's family of optimal fourth-order iterative methods and their dynamics, Fractals 22 (2014) 1450013, doi:10.1142/S0218348X14500133.
[32] B. Neta, M. Scott, C. Chun, Basin attractors for various methods for multiple roots, Appl. Math. Comput. 218 (2012) 5043–5066.
[33] B. Neta, C. Chun, On a family of Laguerre methods to find multiple roots of nonlinear equations, Appl. Math. Comput. 219 (2013) 10987–11004.
[34] B. Neta, C. Chun, Basins of attraction for several optimal fourth order methods for multiple roots, Math. Comput. Simul. 103 (2014) 39–59.
[35] C. Chun, B. Neta, Basins of attraction for Zhou–Chen–Song fourth order family of methods for multiple roots, Math. Comput. Simul. 109 (2015) 74–91.
[36] C. Chun, B. Neta, Comparing the basins of attraction for Kanwar–Bhatia–Kansal family to the best fourth order method, Appl. Math. Comput. 266 (2015) 277–292.
[37] C. Chun, B. Neta, Basin of attraction for several third order methods to find multiple roots of nonlinear equations, Appl. Math. Comput. 268 (2015) 129–137.