2012

Basin attractors for various methods for multiple roots

Neta, Beny; Scott, Melvin; Chun, Changbum

https://hdl.handle.net/10945/39426

Applied Mathematics and Computation 218 (2012) 5043–5066

Contents lists available at SciVerse ScienceDirect

Applied Mathematics and Computation

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

Basin attractors for various methods for multiple roots

Beny Neta [a,*], Melvin Scott [b], Changbum Chun [c]

[a] Department of Applied Mathematics, Naval Postgraduate School, Monterey, CA 93943, USA
[b] 494 Carlton Court, Ocean Isle Beach, NC 28469, USA
[c] Department of Mathematics, Sungkyunkwan University, Suwon 440 746, Republic of Korea

ARTICLE INFO

Keywords:
Nonlinear equations
Multiple roots
Basin of attraction

ABSTRACT

There are several methods for approximating the multiple zeros of a nonlinear function when the multiplicity is known. The methods are classified by the order, informational efficiency and efficiency index. Here we consider other criteria, namely the basin of attraction of the method and its dependence on the order. We discuss all known methods of orders two to four and present the basin of attraction for several examples.

Published by Elsevier Inc.

1. Introduction

In [1], the authors investigated the basin of attraction for several well-known algorithms for the simple roots of a nonlinear equation. The purpose was to propose using the basin of attraction as another method for comparing the algorithms along with such items as order of convergence and efficiency. The authors found that some algorithms have a smooth convergence pattern and others have a rather chaotic pattern, which leads the algorithm to convergence to an unwanted root. In this paper we intend to extend that investigation to algorithms for solving nonlinear equations whose solutions contain roots with multiplicity greater than one.

There is a vast literature on the solution of nonlinear equations and nonlinear systems, see for example Ostrowski [2], Traub [3], Neta [4] and references therein. Here we compare several high order fixed point type methods to approximate a multiple root. Newton's method is only of first order unless it is modified to gain the second order of convergence, see Rall [5] or Schröder [6]. This modification requires a knowledge of the multiplicity. Traub [3] has suggested to use any method for $f^{(m-1)}(x)$ or $f^{1/m}$ or $g(x) = \frac{f(x)}{f'(x)}$.

Any such method will require higher derivatives than the corresponding one for simple zeros. Also the first two of those methods require the knowledge of the multiplicity m. In such a case, there are several other methods developed by Hansen and Patrick [7], Victory and Neta [8], Dong [9,10], Neta and Johnson [11], Neta [12], Li et al. [13], Werner [14] and Neta [15]. Since in general one does not know the multiplicity, Traub [3] suggested a way to approximate it during the iteration. Here we discuss the following methods listed in increasing order of convergence:

Werner: A method of order 1.5 for **double** roots given by Werner [14].

$$\begin{aligned} y_n &= x_n - u_n, \\ x_{n+1} &= x_n - s_n u_n, \end{aligned} \tag{1}$$

* Corresponding author.
E-mail addresses: bneta@nps.edu (B. Neta), MScott8223@atmc.net (M. Scott), cbchun@skku.edu (C. Chun).

0096-3003/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.amc.2011.10.071

where

$$s_n = \begin{cases} \frac{2}{1+\sqrt{1-4f(y_n)/f_n}} & \text{if } f(y_n)/f_n \leqslant \frac{1}{4}, \\ \frac{1}{2} f_n / f(y_n) & \text{otherwise}. \end{cases}$$

We always use

$$u_n = \frac{f_n}{f'_n} \tag{2}$$

and $f_n^{(i)}$ is short for $f^{(i)}(x_n)$.

Newton: The quadratically convergent modified Newton's method is (see Schröder [6])

$$x_{n+1} = x_n - m u_n, \tag{3}$$

N2: A method of order 2.732 (see Neta [15]) requiring the same information as the modified Newton scheme above. The increase in order of convergence is attained by using the derivative at a previous step.

$$x_{n+1} = x_n - \frac{m(m+1)}{2} u_n + \frac{(m-1)^2}{2w(x_n)}, \tag{4}$$

where $w(x_n)$ is given by

$$w(x_n) = \frac{6(f_{n-1} - f_n) + 2hf'_{n-1} + 4hf'_n}{h^2 f'_n}. \tag{5}$$

Halley: The cubically convergent Halley's method [16] which is a special case of the Hansen and Patrick's method [7]

$$x_{n+1} = x_n - \frac{f_n}{\frac{m+1}{2m} f'_n - \frac{f_n f''_n}{2f'_n}}. \tag{6}$$

VN: The third order method developed by Victory and Neta [8]

$$\begin{aligned} y_n &= x_n - u_n, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'_n} \frac{f_n + Af(y_n)}{f_n + Bf(y_n)}, \end{aligned} \tag{7}$$

where

$$\begin{aligned} A &= \mu^{2m} - \mu^{m+1}, \\ B &= -\frac{\mu^m (m-2)(m-1) + 1}{(m-1)^2}, \\ \mu &= \frac{m}{m-1}. \end{aligned}$$

NC: The third order method developed by Neta [15] and based on Chebyshev's method (see [17–19]).

$$\begin{aligned} y_n &= x_n - \alpha u_n, \\ x_{n+1} &= x_n - u_n \left[\beta + \gamma \frac{f(y_n)}{f_n} \right], \end{aligned} \tag{8}$$

where

$$\begin{aligned} \alpha &= \frac{1}{2} \frac{m(m+3)}{m+1}, \\ \beta &= \frac{m^3 + 4m^2 + 9m + 2}{(m+3)^2}, \\ \gamma &= \frac{2^{m+1}(m^2-1)}{(m+3)^2 \left(\frac{m-1}{m+1}\right)^m}. \end{aligned}$$

D: The four third order methods developed by Dong [9,10]:

D1:

$$\begin{aligned} y_n &= x_n - \sqrt{m}\, u_n, \\ x_{n+1} &= y_n - m \left(1 - \frac{1}{\sqrt{m}} \right)^{1-m} \frac{f(y_n)}{f'_n}, \end{aligned} \tag{9}$$

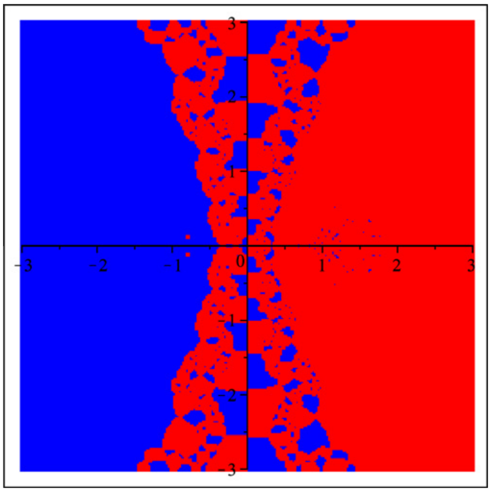

Fig. 1. Werner's method for the polynomial whose roots are both double: −1, 1.

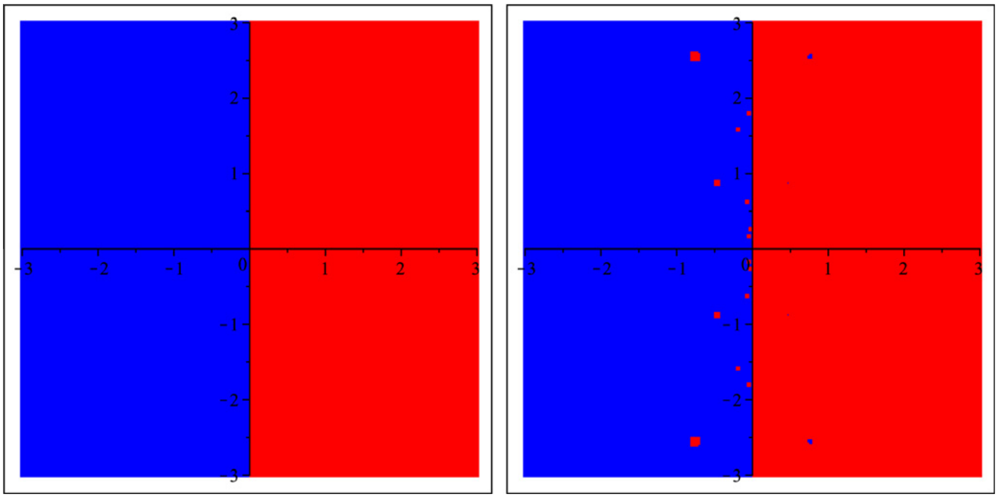

Fig. 2. Newton's method (left) and Neta's N2 method (right) for the polynomial whose roots are both double: −1, 1.

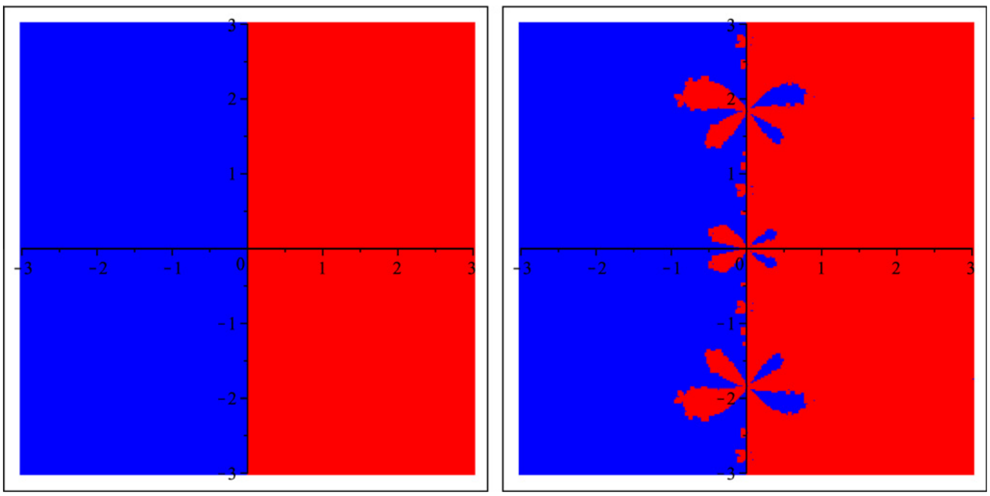

Fig. 3. Third order method due to Halley (left) and Neta's NC scheme (right) for the polynomial whose roots are both double: −1, 1.

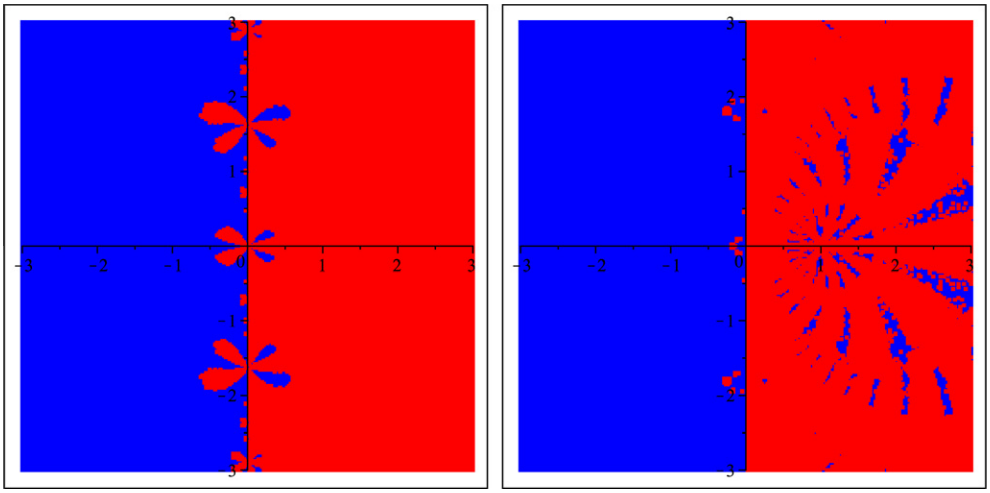

Fig. 4. Dong's D1 third order (left) and Dong's D2 third order (right) for the polynomial whose roots are both double: −1, 1.

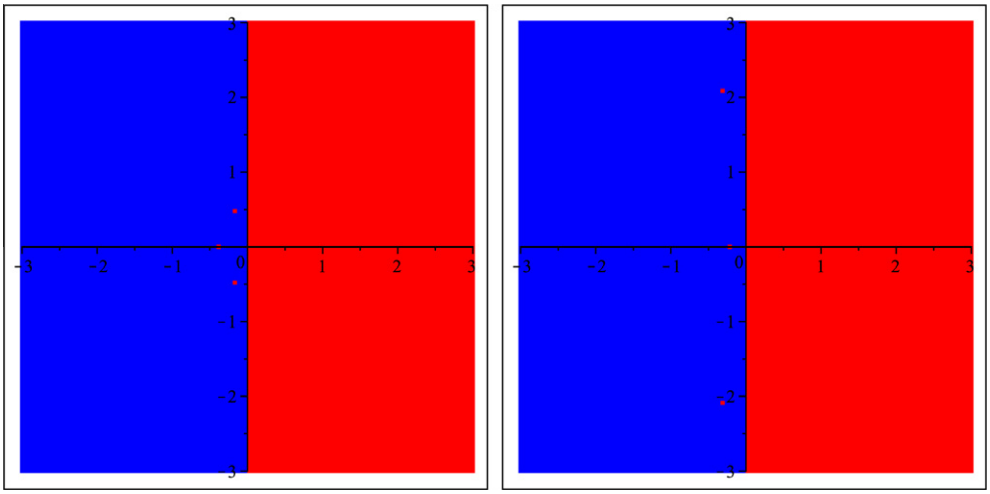

Fig. 5. Dong's D3 third order (left) and Dong's D4 third order (right) for the polynomial whose roots are both double: −1, 1.

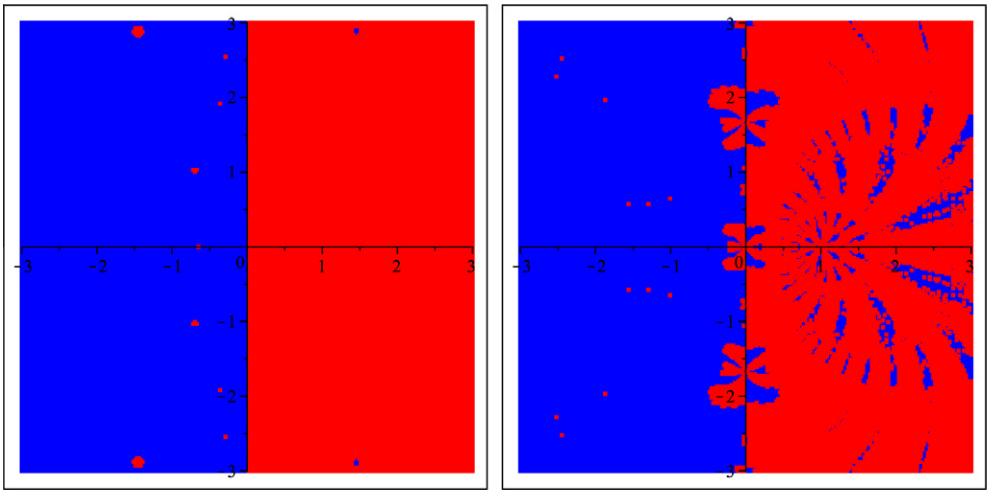

Fig. 6. Osada's third order method (left) and Victory and Neta (right) for the polynomial whose roots are both double: −1, 1.

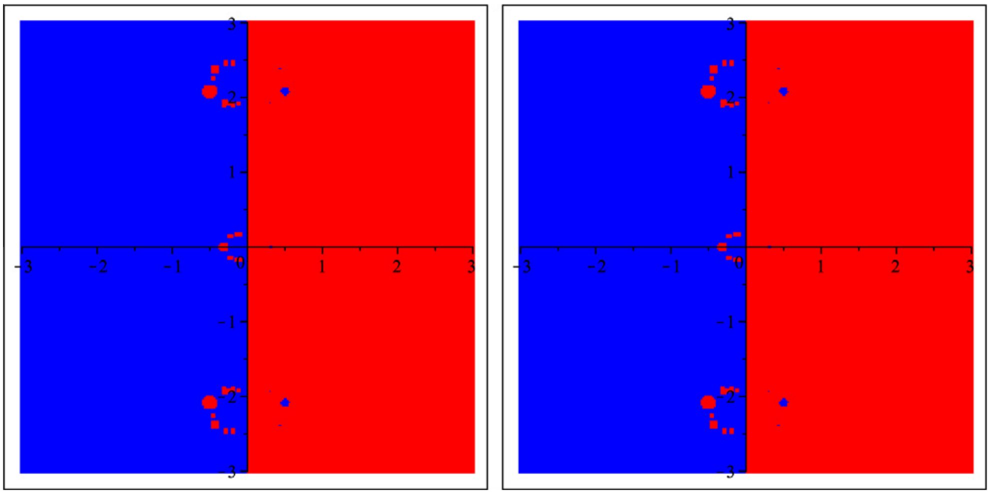

Fig. 7. LCN1 method of order four (left) and LCN2 method of order four (right) for the polynomial whose roots are both double: −1, 1.

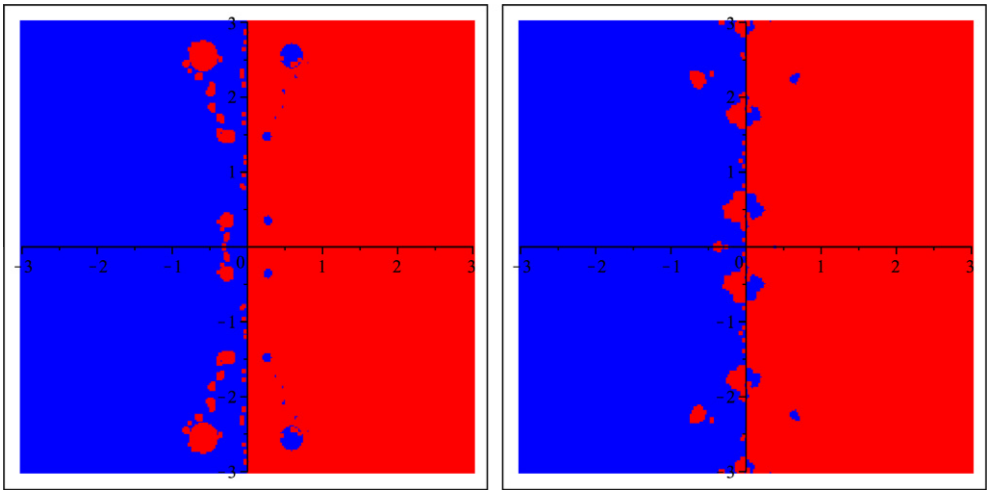

Fig. 8. LCN3 method of order four (left) and LCN4 method of order four (right) for the polynomial whose roots are both double: −1, 1.

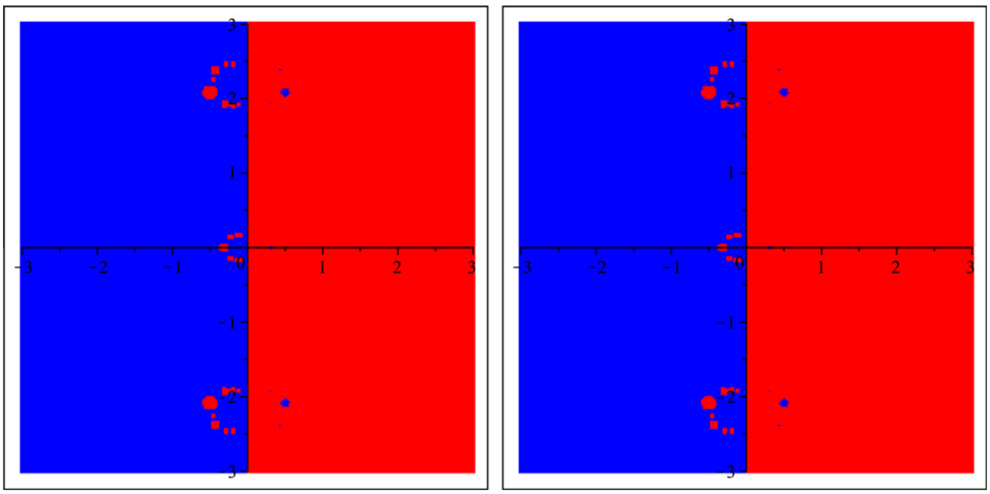

Fig. 9. LCN5 method of order four (left) and LCN6 method of order four (right) for the polynomial whose roots are both double: −1, 1.

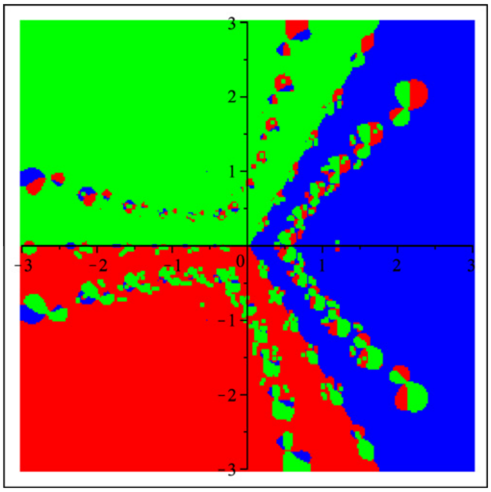

Fig. 10. Werner's method for the polynomial $(z^3 - 1)^2$.

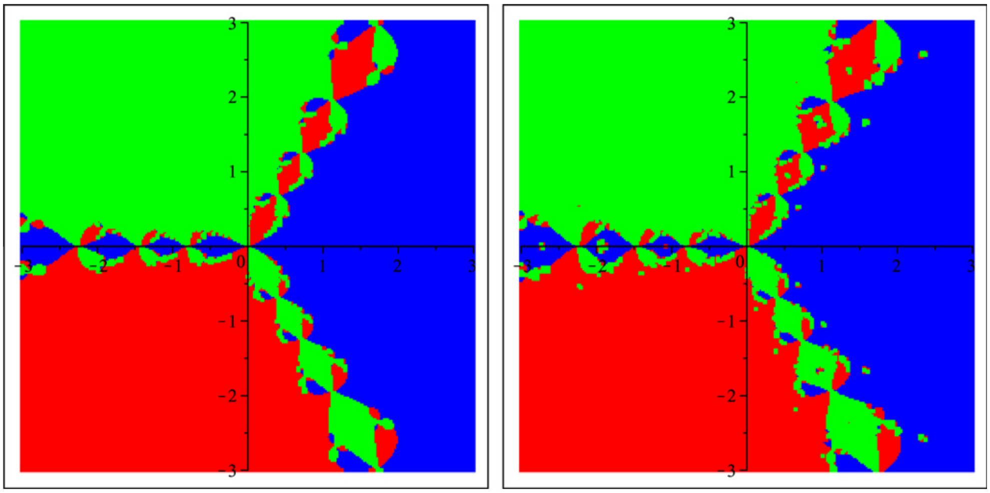

Fig. 11. Newton's method (left) and Neta's N2 method (right) for the polynomial $(z^3 - 1)^2$.

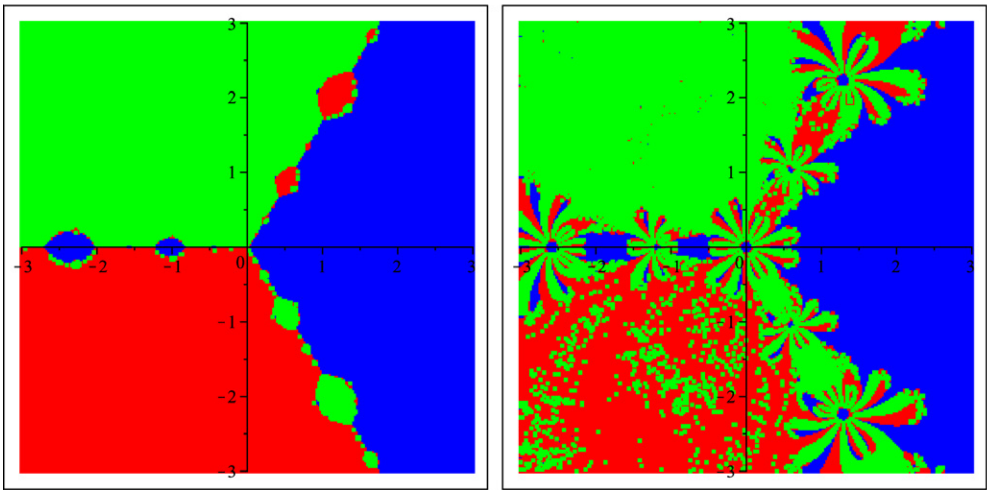

Fig. 12. Third order method due to Halley (left) and Neta's NC scheme (right) for the polynomial $(z^3 - 1)^2$.

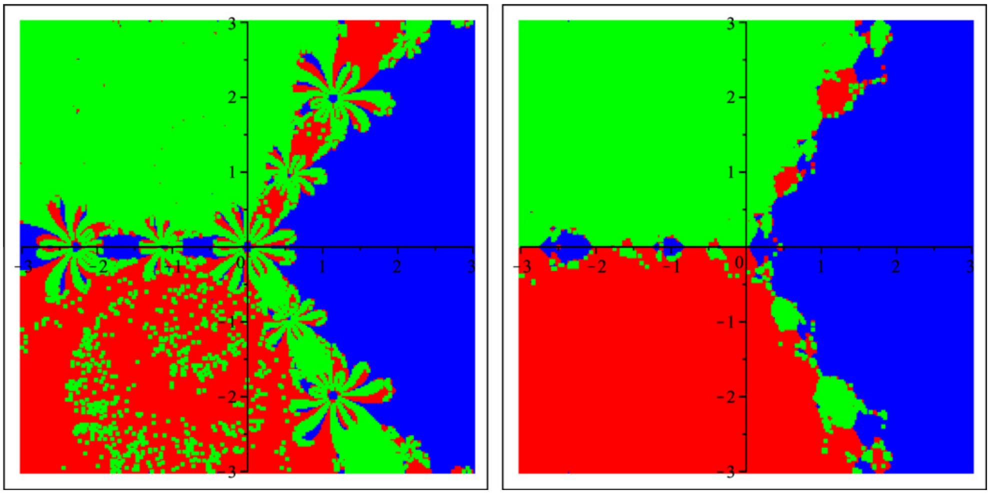

Fig. 13. Dong's D1 third order (left) and Dong's D2 third order (right) for the polynomial $(z^3 - 1)^2$.

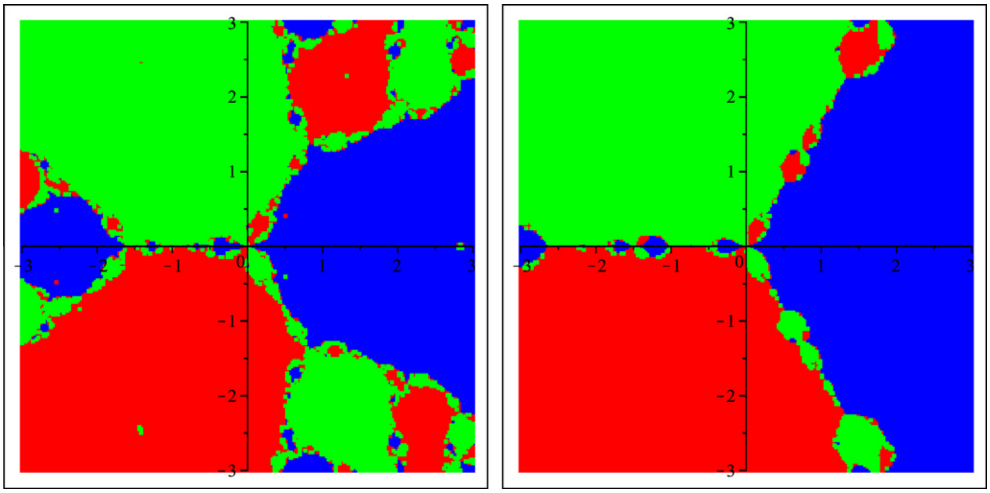

Fig. 14. Dong's D3 third order (left) and Dong's D4 third order (right) for the polynomial $(z^3 - 1)^2$.

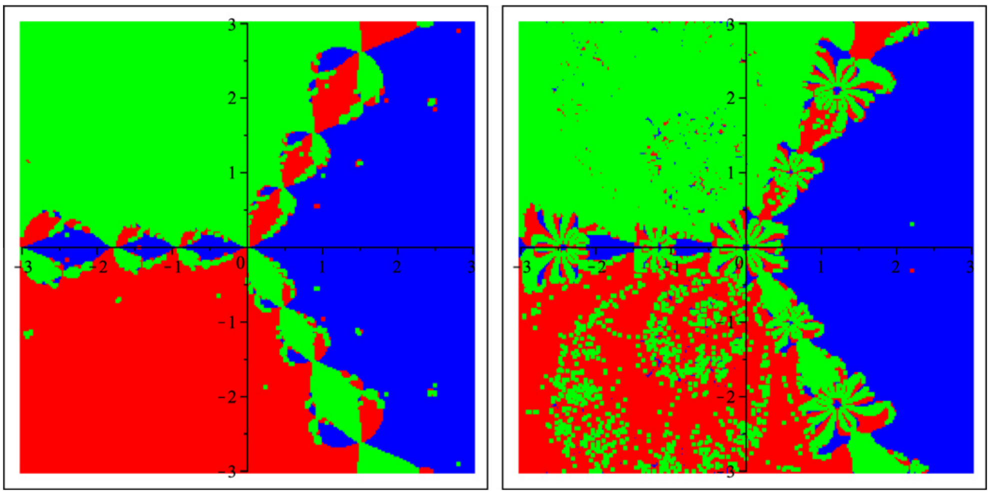

Fig. 15. Osada's third order method (left) and Victory and Neta (right) for the polynomial $(z^3 - 1)^2$.

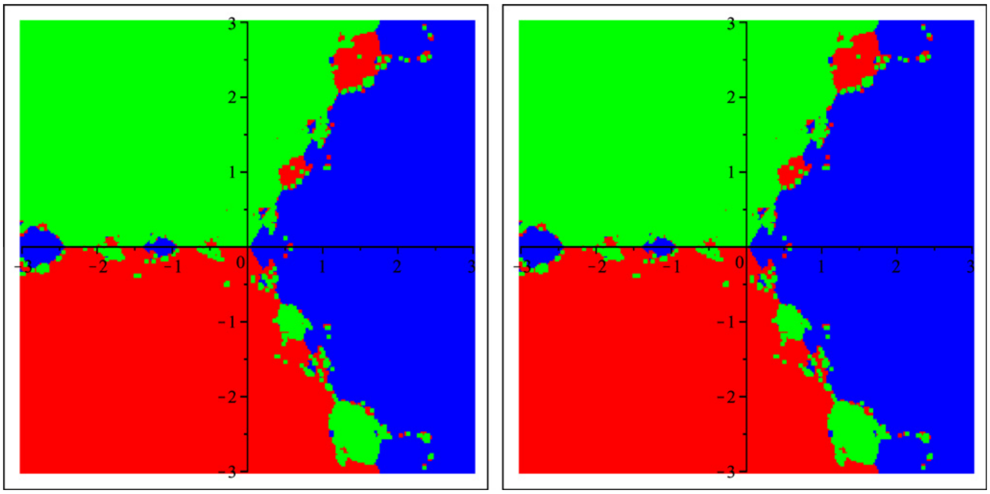

Fig. 16. LCN1 method of order four (left) and LCN2 method of order four (right) for the polynomial $(z^3 - 1)^2$.

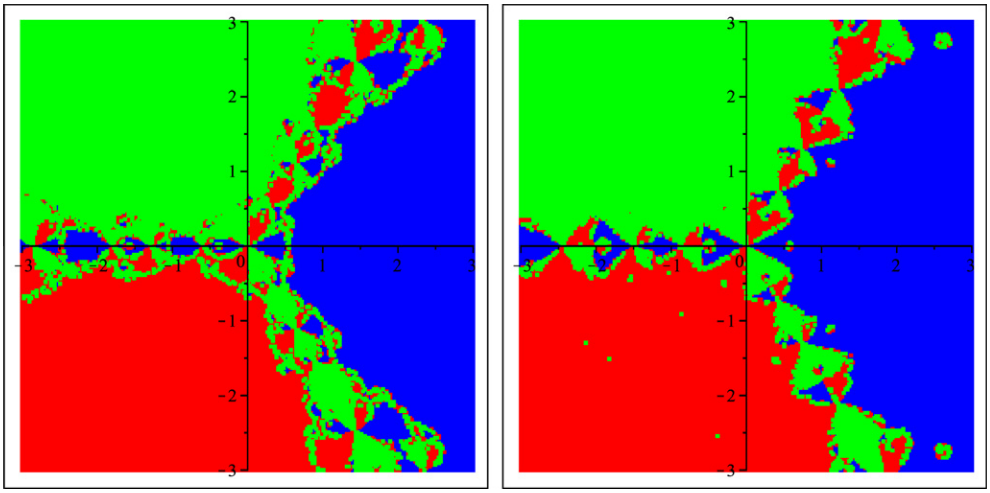

Fig. 17. LCN3 method of order four (left) and LCN4 method of order four (right) for the polynomial $(z^3 - 1)^2$.

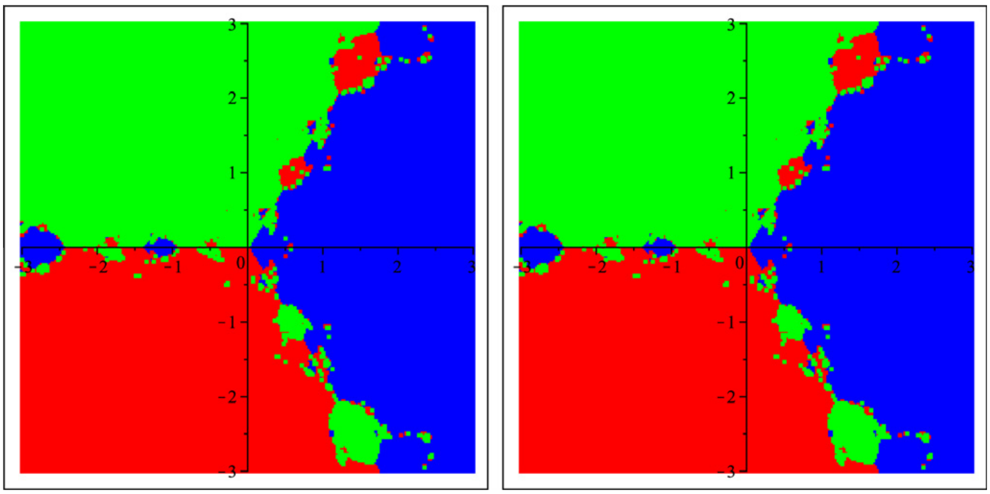

Fig. 18. LCN5 method of order four (left) and LCN6 method of order four (right) for the polynomial $(z^3 - 1)^2$.

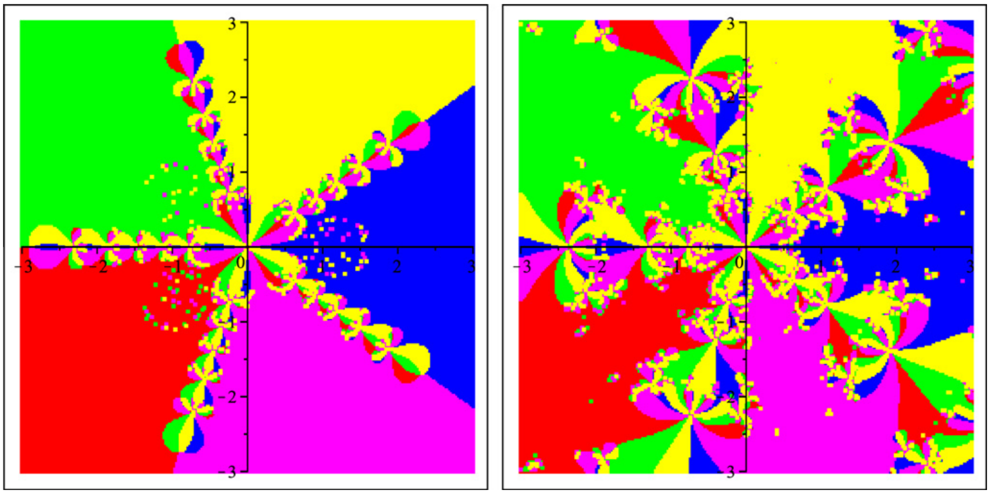

Fig. 19. Newton's method (left) and Neta's N2 method (right) for the polynomial $(z^5 - 1)^3$.

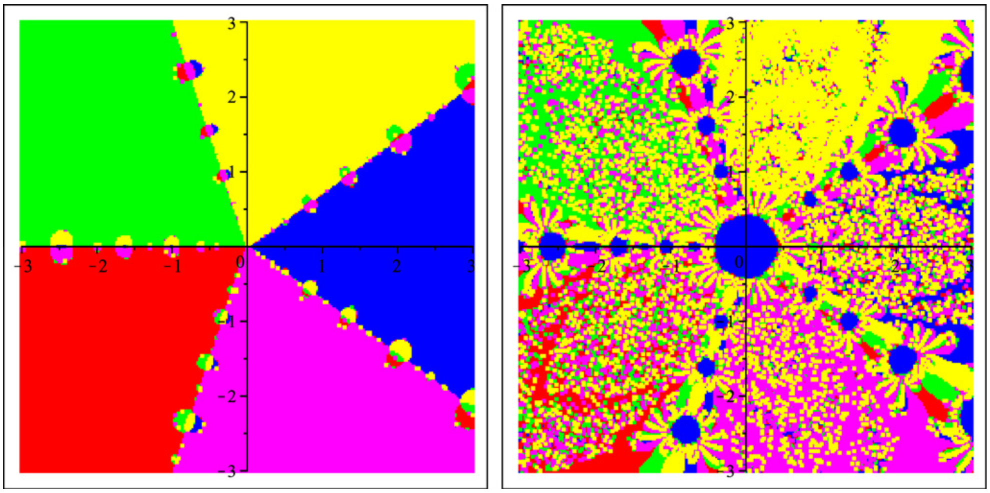

Fig. 20. Third order method due to Halley (left) and Neta's NC scheme (right) for the polynomial $(z^5 - 1)^3$.

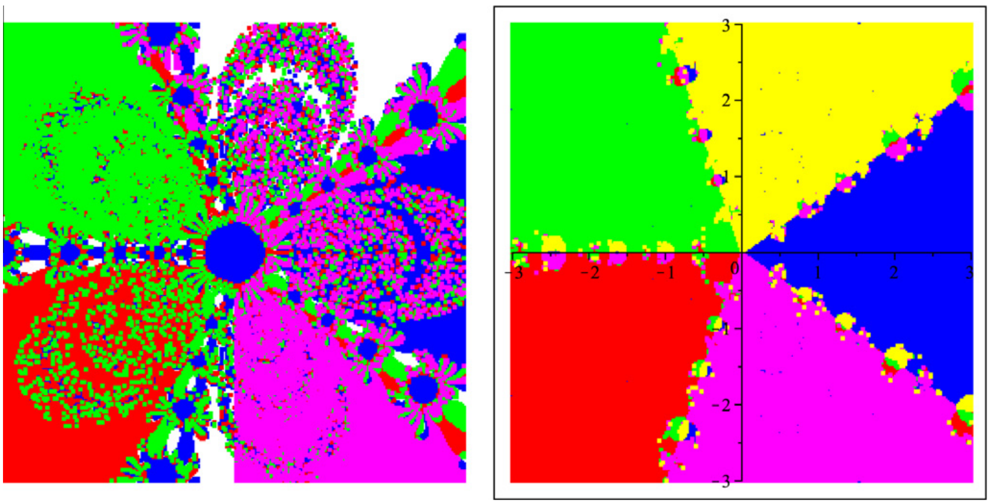

Fig. 21. Dong's D1 third order (left) and Dong's D2 third order (right) for the polynomial $(z^5 - 1)^3$.

D2:

$$
\begin{aligned}
y_n &= x_n - u_n, \\
x_{n+1} &= y_n + \frac{u_n\, f(y_n)}{f(y_n) - \left(1 - \frac{1}{m}\right)^{m-1} f_n},
\end{aligned}
\tag{10}
$$

D3:

$$
\begin{aligned}
y_n &= x_n - u_n, \\
x_{n+1} &= y_n - \frac{f_n}{\left(\frac{m}{m-1}\right)^{m+1} f'(y_n) + \frac{m-m^2-1}{(m-1)^2} f'_n},
\end{aligned}
\tag{11}
$$

D4:

$$
\begin{aligned}
y_n &= x_n - \frac{m}{m+1} u_n, \\
x_{n+1} &= y_n - \frac{\frac{m}{m+1} f_n}{\left(1 + \frac{1}{m}\right)^m f'(y_n) - f'_n}.
\end{aligned}
\tag{12}
$$

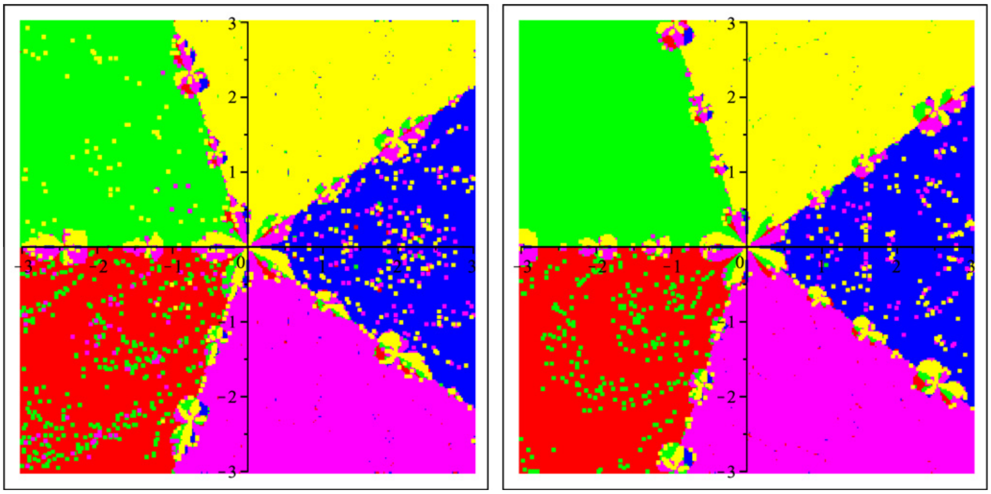

Fig. 22. Dong's D3 third order (left) and Dong's D4 third order (right) for the polynomial $(z^5 - 1)^3$.

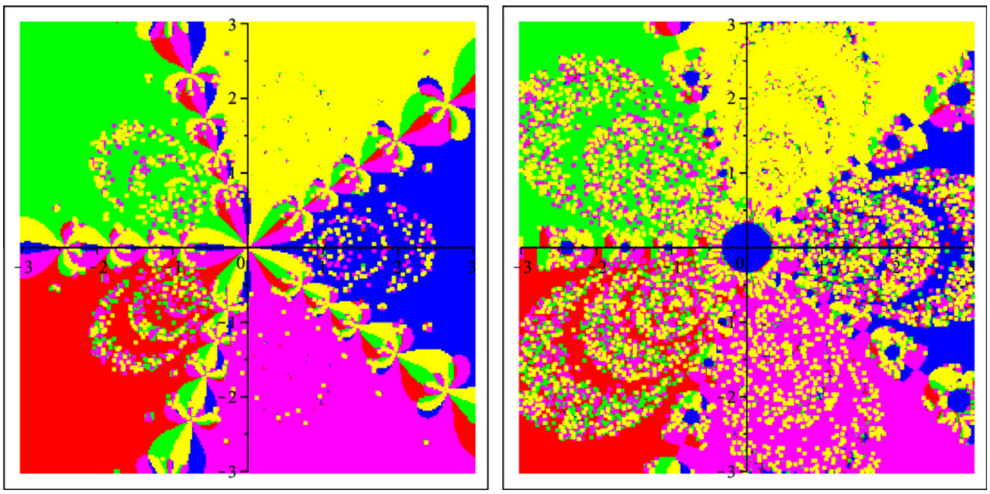

Fig. 23. Osada's third order method (left) and Victory and Neta (right) for the polynomial $(z^5 - 1)^3$.

Osada: The third order method due to Osada [20]

$$x_{n+1} = x_n - \frac{1}{2}m(m+1)u_n + \frac{1}{2}(m-1)^2\frac{f'_n}{f''_n}. \tag{13}$$

LCN: The six fourth order methods developed by Li et al. [13] and based on the results of Neta and Johnson [11] and Neta [12].

LCN1:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ \eta_n &= x_n - \frac{2m}{m+2}u_n + 2\left(\frac{m}{m+2}\right)^m v_n, \\ x_{n+1} &= x_n - \frac{f(x_n)}{a_1 f'(x_n) + a_2 f'(y_n) + a_3 f'(\eta_n)}, \end{aligned} \tag{14}$$

where we always use

$$v_n = \frac{f_n}{f'(y_n)} \tag{15}$$

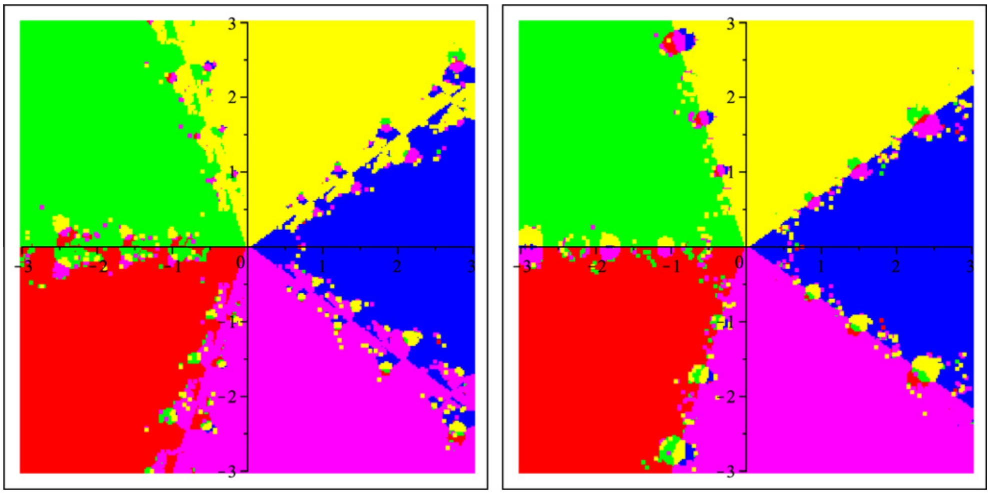

Fig. 24. LCN1 method of order four (left) and LCN2 method of order four (right) for the polynomial $(z^5-1)^3$.

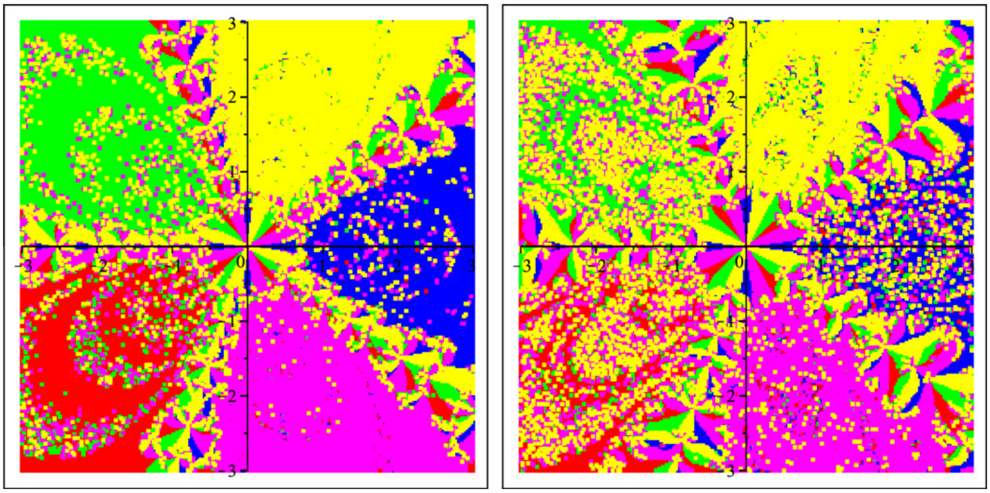

Fig. 25. LCN3 method of order four (left) and LCN4 method of order four (right) for the polynomial $(z^5-1)^3$.

and

$$a_1 = -\frac{1}{16}\frac{3m^4+16m^3+40m^2-176}{m(m+8)},$$
$$a_2 = \frac{1}{8}\frac{m^4+3m^3+10m^2-4m+8}{\left(\frac{m}{m+2}\right)^m m(m+8)},$$
$$a_3 = \frac{1}{16}\frac{m^5+6m^4+8m^3-16m^2-48m-32}{m^2(m+8)}.$$

LCN2:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ \eta_n &= x_n - 2\left(\frac{m}{m+2}\right)^m v_n, \\ x_{n+1} &= x_n - \frac{f(x_n)}{a_1 f'(x_n) + a_2 f'(y_n) + a_3 f'(\eta_n)}, \end{aligned} \tag{16}$$

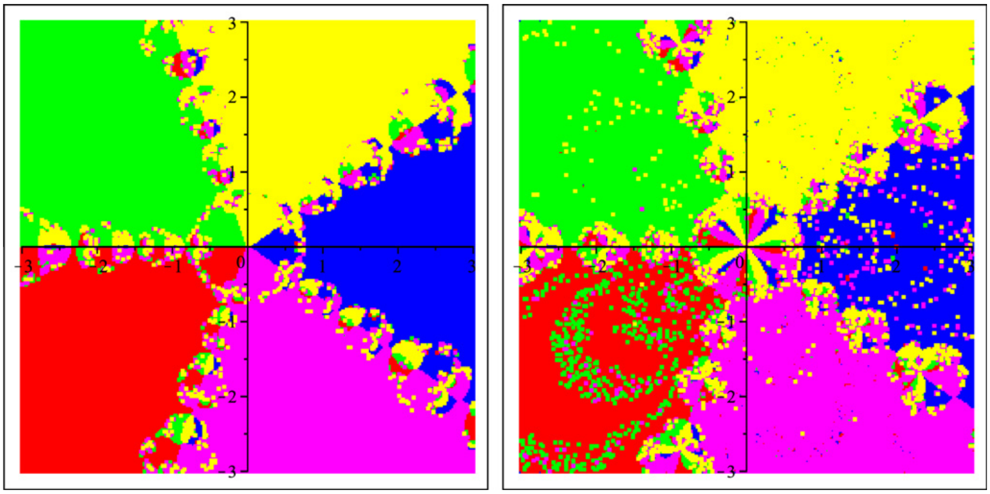

Fig. 26. LCN5 method of order four (left) and LCN6 method of order four (right) for the polynomial $(z^5-1)^3$.

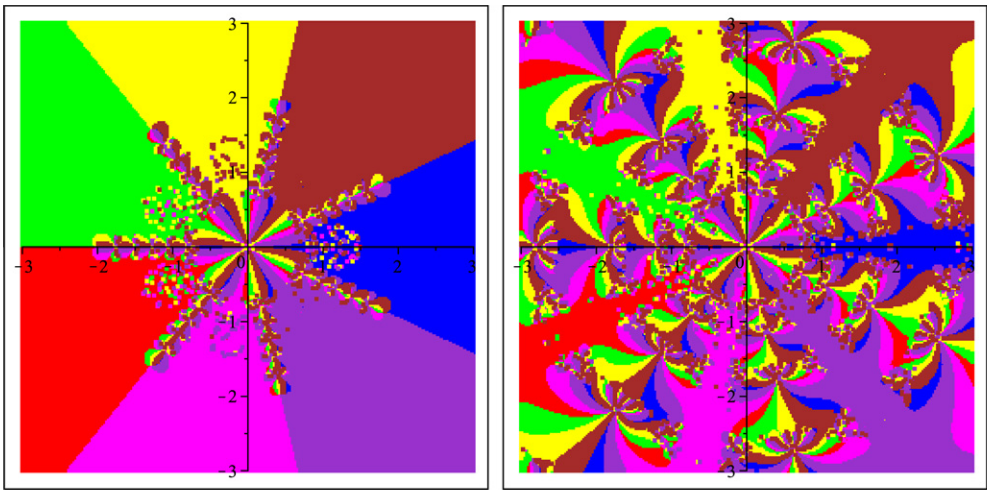

Fig. 27. Newton's method (left) and Neta's N2 method (right) for the polynomial $(z^7-1)^4$.

where

$$a_1 = \frac{1}{8}\,\frac{m^6-m^5-14m^4+12m^3+48m^2-80m+32}{m(m^3+2m^2-8m+4)},$$

$$a_2 = -\frac{m}{16}\,\frac{3m^4-6m^3-20m^2+40m-16}{\left(\frac{m}{m+2}\right)^m(m^3+2m^2-8m+4)},$$

$$a_3 = \frac{1}{16}\,\frac{m^3(m^2-4)}{\left(\frac{m}{m+2}\right)^m(m^3+2m^2-8m+4)}.$$

LCN3:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ \eta_n &= x_n - \frac{2m}{m+2}u_n + 2\left(\frac{m}{m+2}\right)^m v_n, \\ x_{n+1} &= x_n - a_1 u_n - a_2 v_n - a_3\frac{f(x_n)}{f'(\eta_n)}, \end{aligned} \tag{17}$$

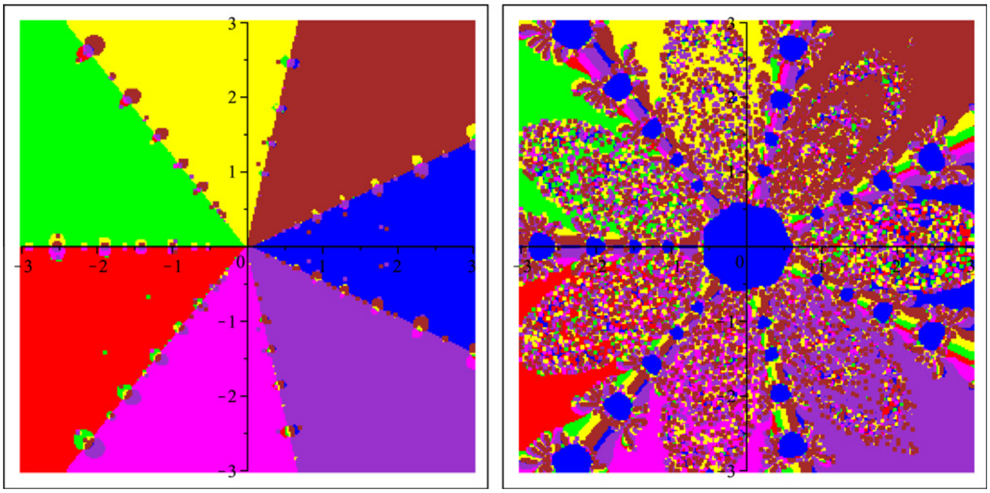

Fig. 28. Third order method due to Halley (left) and Neta's NC scheme (right) for the polynomial $(z^7-1)^4$.

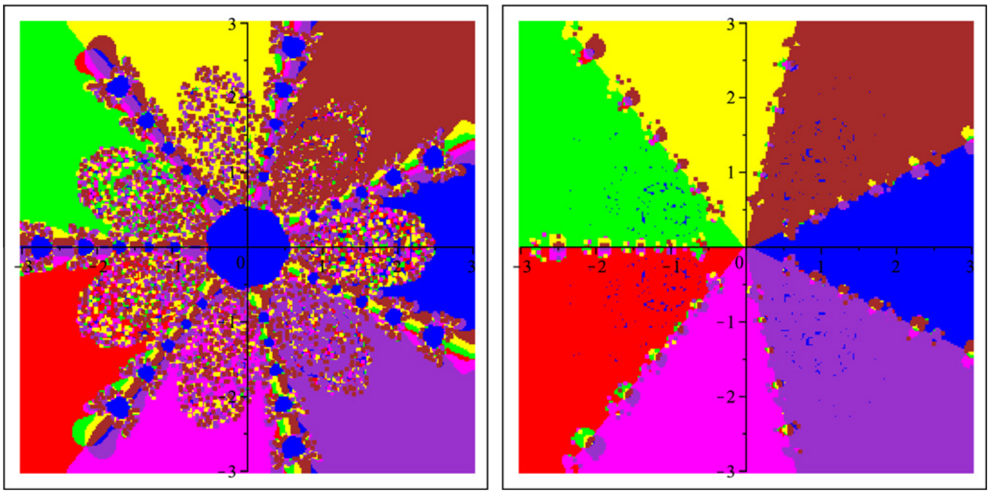

Fig. 29. Dong's D1 third order (left) and Dong's D2 third order (right) for the polynomial $(z^7-1)^4$.

where

$$a_1 = \frac{m}{8}\frac{m^4+4m^3-8m+48}{m^2+2m+6},$$
$$a_2 = \frac{1}{4}\frac{\left(\frac{m}{m+2}\right)^m m(m^3+12m^2+36m+32)}{m^2+2m+6},$$
$$a_3 = -\frac{1}{8}\frac{m^2(m^3+6m^2+12m+8)}{m^2+2m+6}.$$

LCN4:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ \eta_n &= x_n - 2\left(\frac{m}{m+2}\right)^m v_n, \\ x_{n+1} &= x_n - a_1 u_n - a_2 v_n - a_3\frac{f(x_n)}{f'(\eta_n)}, \end{aligned} \tag{18}$$

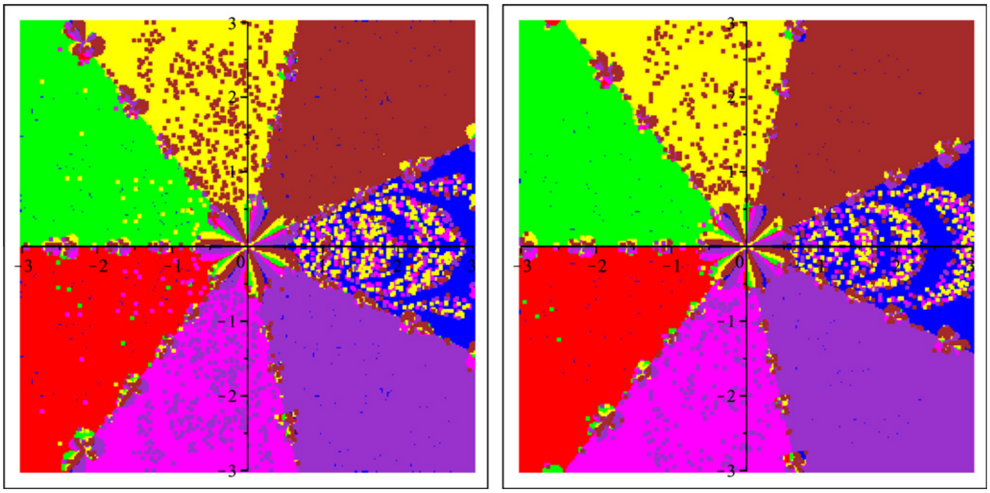

Fig. 30. Dong's D3 third order (left) and Dong's D4 third order (right) for the polynomial $(z^7-1)^4$.

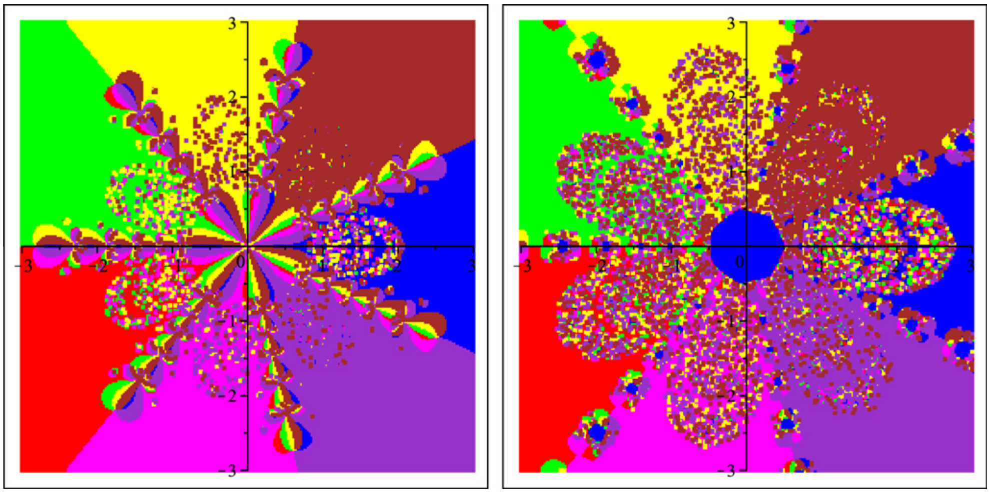

Fig. 31. Osada's third order method (left) and Victory and Neta (right) for the polynomial $(z^7-1)^4$.

where

$$a_1 = -\frac{1}{4}\,\frac{m(2m^4-m^3-12m^2+20m-8)}{m^2-4m+2},$$
$$a_2 = \frac{1}{8}\,\frac{\left(\frac{m}{m+2}\right)^m m(5m^4+10m^3-16m^2-24m+16)}{m^2-4m+2},$$
$$a_3 = -\frac{1}{8}\,\frac{m^3(m+2)^2\left(\frac{m}{m+2}\right)^m}{m^2-4m+2}.$$

LCN5:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ x_{n+1} &= x_n - a_3\frac{f(x_n)}{f'(y_n)} - \frac{f(x_n)}{b_1 f'(x_n) + b_2 f'(y_n)}, \end{aligned} \tag{19}$$

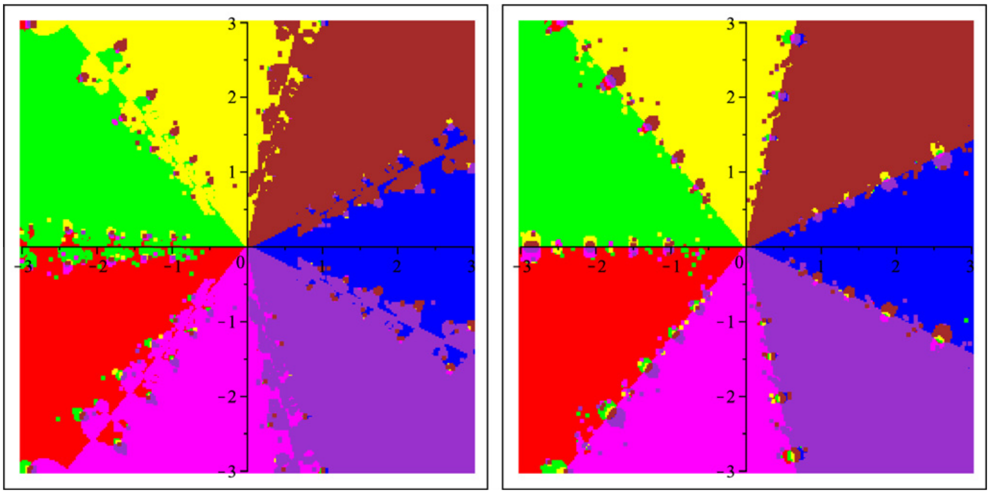

Fig. 32. LCN1 method of order four (left) and LCN2 method of order four (right) for the polynomial $(z^7-1)^4$.

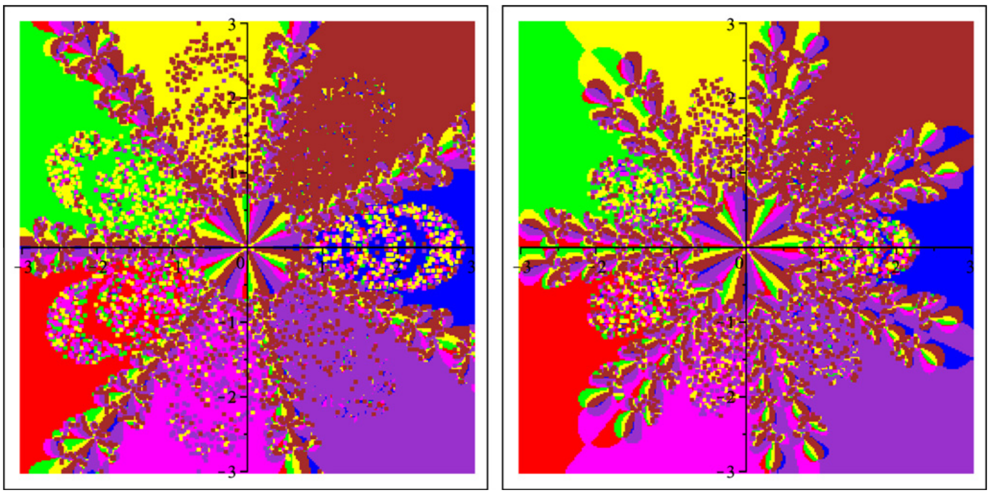

Fig. 33. LCN3 method of order four (left) and LCN4 method of order four (right) for the polynomial $(z^7-1)^4$.

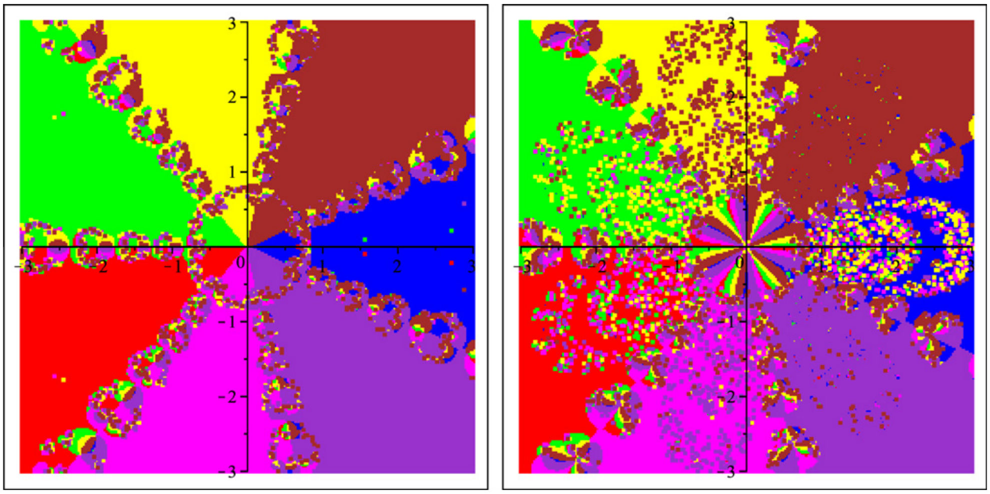

Fig. 34. LCN5 method of order four (left) and LCN6 method of order four (right) for the polynomial $(z^7 - 1)^4$.

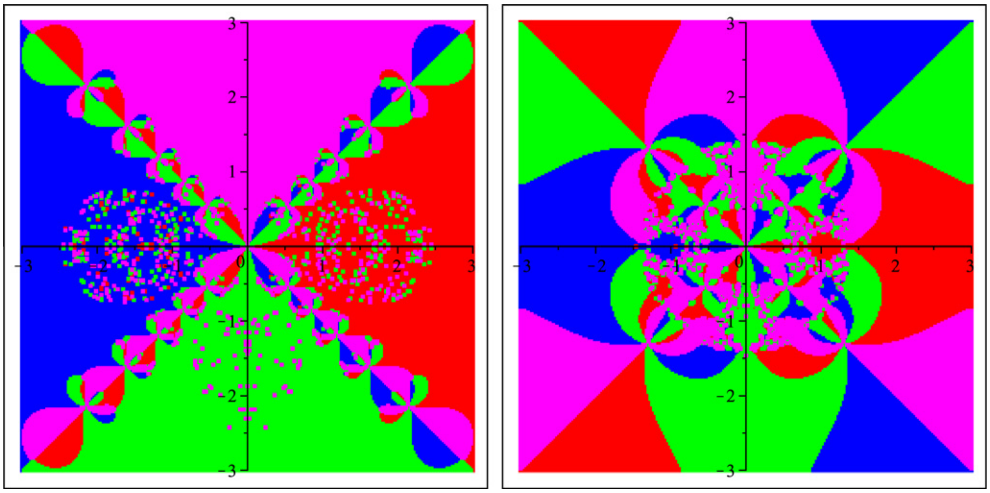

Fig. 35. Newton's method (left) and Neta's N2 method (right) for the polynomial $(z^4 - 1)^5$.

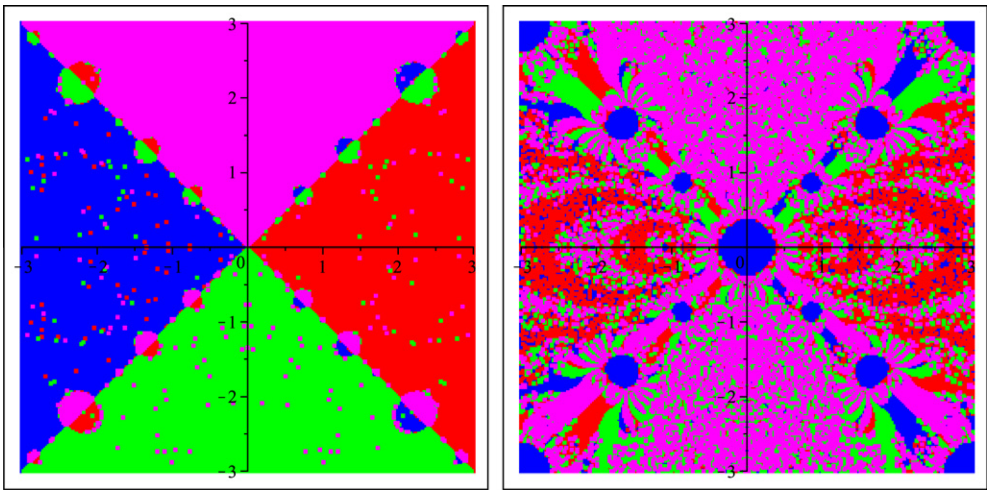

Fig. 36. Third order method due to Halley (left) and Neta's NC scheme (right) for the polynomial $(z^4 - 1)^5$.

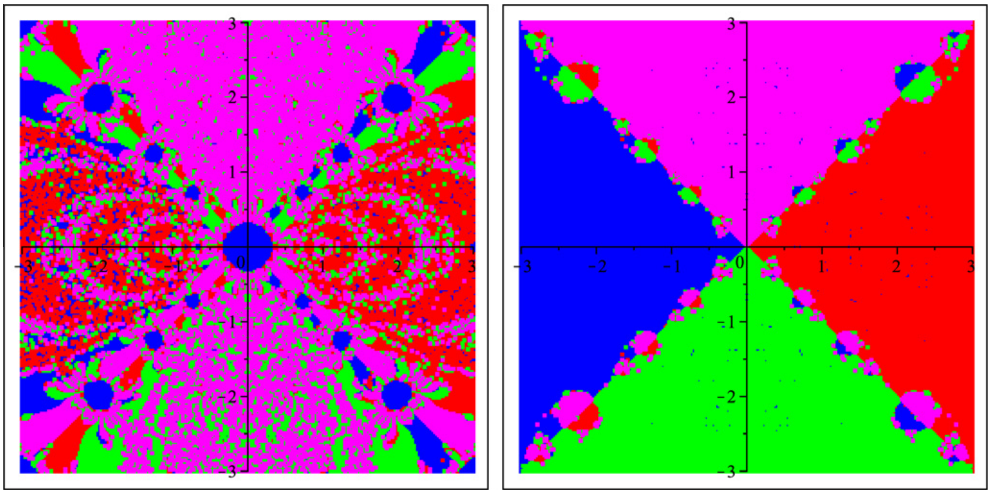

Fig. 37. Dong's D1 third order (left) and Dong's D2 third order (right) for the polynomial $(z^4 - 1)^5$.

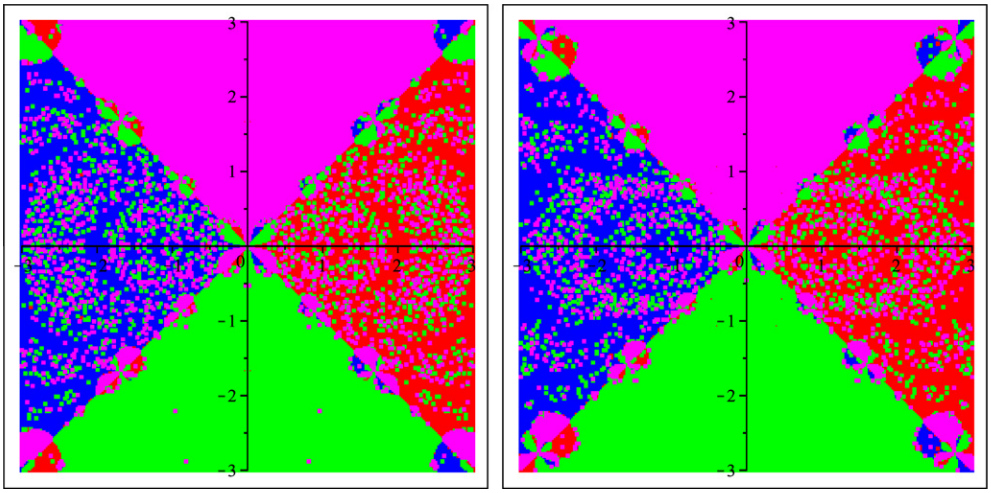

Fig. 38. Dong's D3 third order (left) and Dong's D4 third order (right) for the polynomial $(z^4 - 1)^5$.

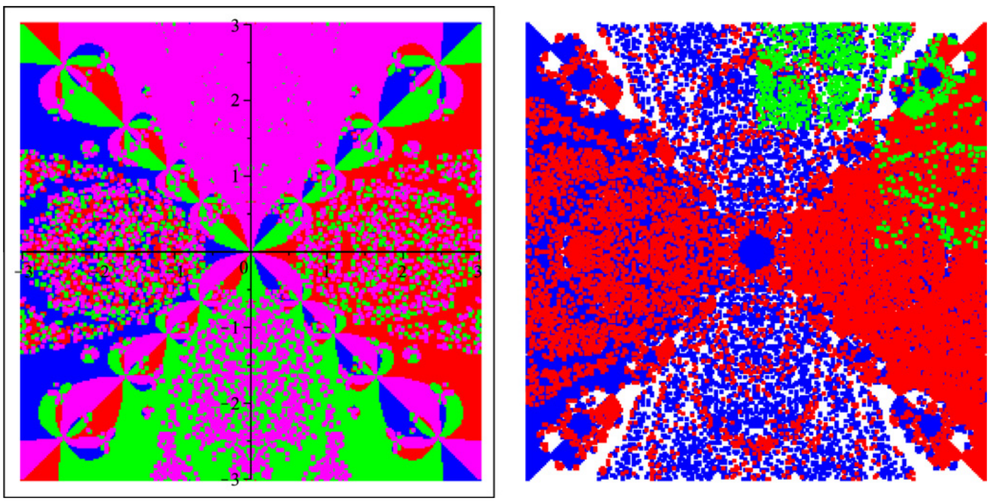

Fig. 39. Osada's third order method (left) and Victory and Neta (right) for the polynomial $(z^4 - 1)^5$.

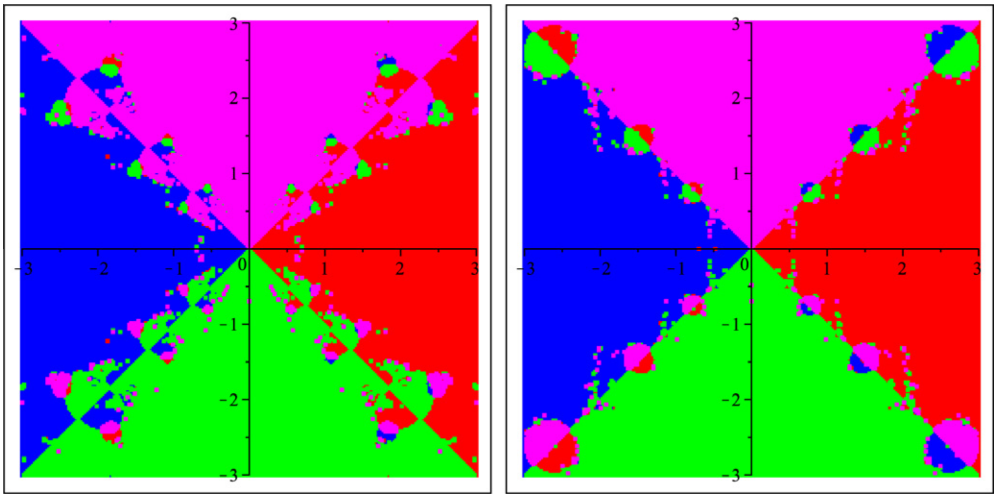

Fig. 40. LCN1 method of order four (left) and LCN2 method of order four (right) for the polynomial $(z^4 - 1)^5$.

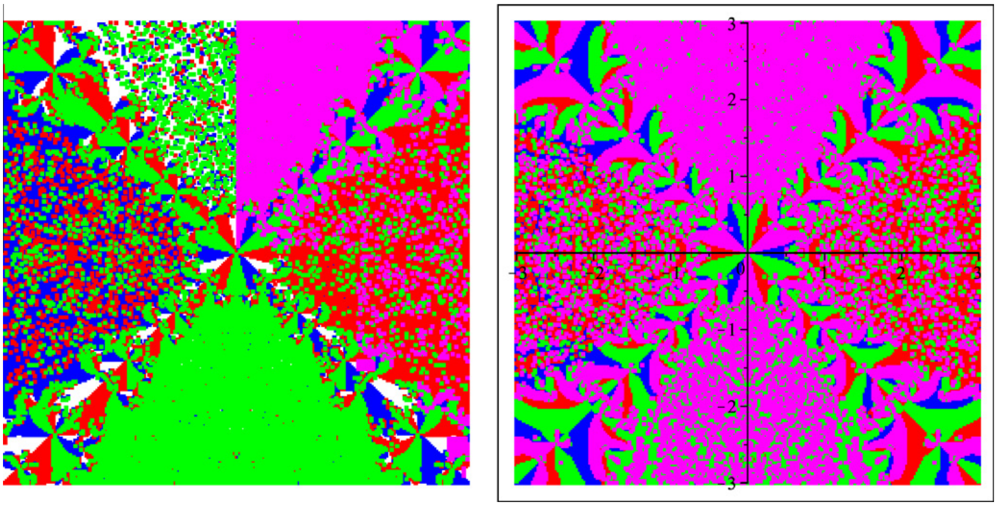

Fig. 41. LCN3 method of order four (left) and LCN4 method of order four (right) for the polynomial $(z^4 - 1)^5$.

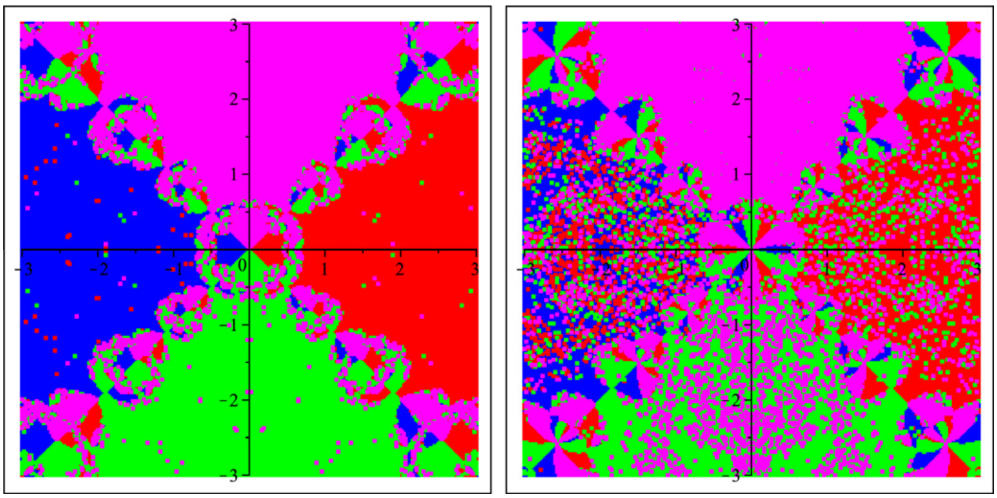

Fig. 42. LCN5 method of order four (left) and LCN6 method of order four (right) for the polynomial $(z^4 - 1)^5$.

where

$$a_3 = -\frac{1}{2}\frac{\left(\frac{m}{m+2}\right)^m m(m^4+4m^3-16m-16)}{m^3-4m+8},$$
$$b_1 = -\frac{(m^3-4m+8)^2}{m(m^4+4m^3-4m^2-16m+16)(m^2+2m-4)},$$
$$b_2 = \frac{m^2(m^3-4m+8)}{\left(\frac{m}{m+2}\right)^m (m^4+4m^3-4m^2-16m+16)(m^2+2m-4)}.$$

LCN6:

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}u_n, \\ x_{n+1} &= x_n - a_3\frac{f(x_n)}{f'(x_n)} - \frac{f(x_n)}{b_1 f'(x_n) + b_2 f'(y_n)}, \end{aligned} \tag{20}$$

where

$$a_3 = -\frac{1}{2}m^2 + m,$$
$$b_1 = -\frac{1}{m}, \quad b_2 = \frac{1}{m\left(\frac{m}{m+2}\right)^m}.$$

The Basin of Attraction is a method to visually comprehend how an algorithm behaves as a function of the various starting points. Natural questions then are:

Table 1
Comparison of various methods.

Method	Example 1	Example 2	Example 3	Example 4	Example 5	Average
Werner	4	2	–	–	–	3
Newton	1	1	2	2	3	9/5
N2	2	1	3	3	4	13/5
Halley	1	1	1	1	3	7/5
NC	3	3	4	4	4	18/5
D1	3	3	4	4	4	18/5
D2	4	1	1	2	1	9/5
D3	1	2	4	4	3	14/5
D4	1	1	4	4	3	13/5
Osada	2	1	4	4	4	3
VN	4	3	4	4	4	19/5
LCN1	3	1	2	2	2	2
LCN2	3	1	1	1	1	7/5
LCN3	3	1	4	4	3	3
LCN4	3	1	4	4	4	16/5
LCN5	3	1	3	3	4	14/5
LCN6	3	1	4	4	4	16/5

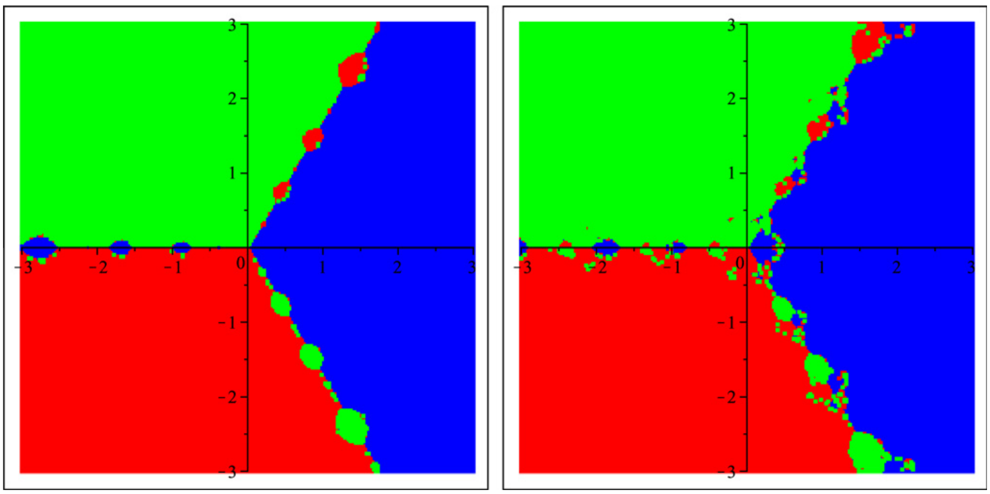

Fig. 43. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^3-1)^3$.

- How does the Basin of Attraction differ for algorithms with the same order of convergence?
- How does the Basin of Attraction differ for algorithms with different order of convergence?
- Can the differences be used to compare various algorithms?

In this paper we will discuss some qualitative issues using the basin of attraction as a criterion for comparison. This idea has been used by the authors to compare methods for approximating simple zeros (see [1].)

2. Numerical experiments

We have used the above methods for 5 different polynomials having multiple roots with multiplicity $m = 2, 3, 4, 5$. Clearly Werner's method is only for double roots and it is used only in the first two examples for which $m = 2$.

In our first example, we have taken the polynomial

$$p_1(z) = (z-1)^2(z+1)^2, \tag{21}$$

whose roots $z = \pm 1$ are both real and of multiplicity $m = 2$. Based on Figs. 1–9 we can see that Newton's method (Fig. 2 left), Halley's method (Fig. 3 left) and Dong's D3 and D4 schemes (Fig. 5) are best, followed by Neta's N2 (Fig. 2 right) and Osada's method (Fig. 6 left). All other methods are not competitive, some will converge to $z = -1$ even in the neighborhood of the other root $z = 1$. See, for example, Dong's D2 (Fig. 4 right) and Victory-Neta (Fig. 6 right) where we see many blue dots on

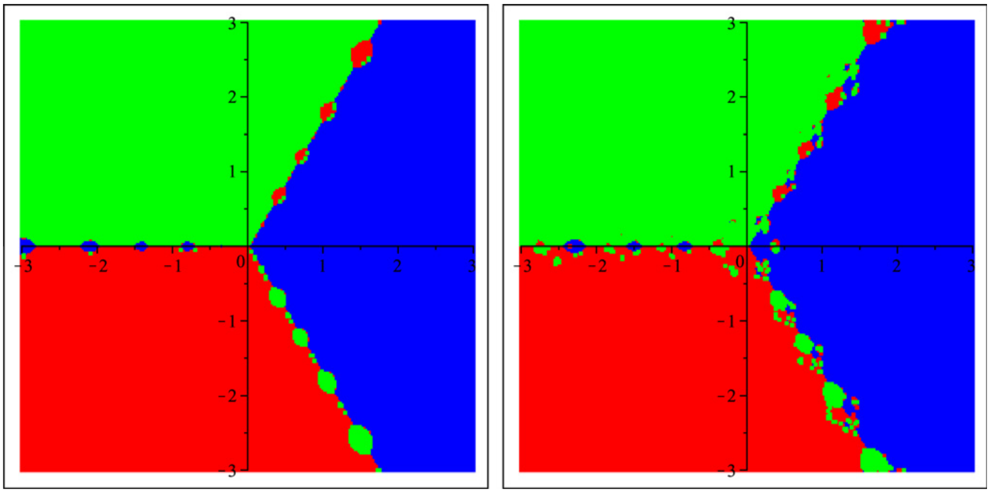

Fig. 44. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^3-1)^4$.

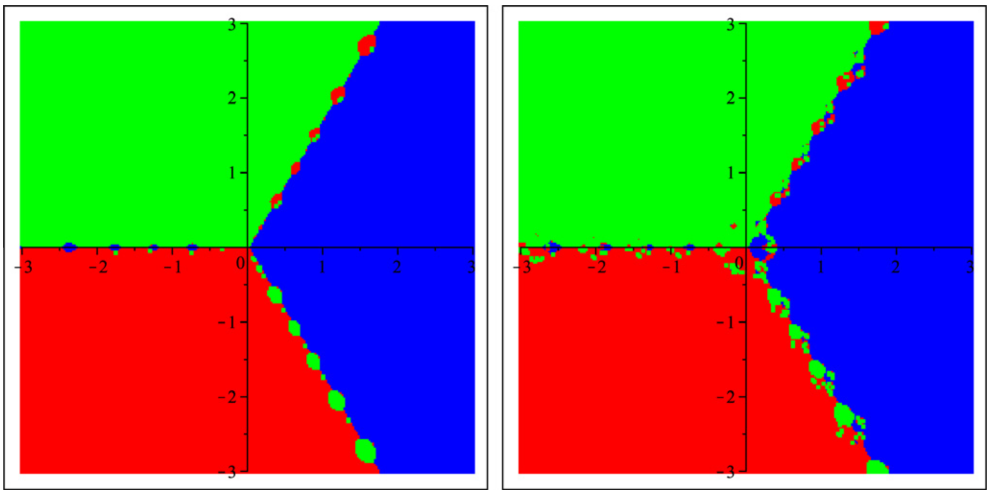

Fig. 45. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^3-1)^5$.

the red side. This means that there are many points on the right that converge to the root on the left. The worst method is Werner's scheme (Fig. 1).

Our next example is also having double roots. The polynomial have the three roots of unity,

$$p_2(z) = (z^3 - 1)^2. \tag{22}$$

The results are presented in Figs. 10–18. Again Newton's method (Fig. 11 left) performs very well. Halley's method (Fig. 12 left) performed better than Newton's. Neta's N2 scheme (Fig. 11 right) shows similar results to Newton's. Two of Dong's schemes, D2 (Figs. 13 right) and D4 (Fig. 14 right) and LCN1 (Fig. 16 left), LCN2 (Fig. 16 right), LCN4 (Fig. 17 right), LCN5 (Fig. 18 left) and LCN6 methods (Fig. 18 right) show good results. The rest are not as good. In fact the method NC (Fig. 12 right), Dong's D1 (Fig. 13 left) and Victory-Neta scheme (Fig. 15 right) show convergence to the wrong root, see the neighborhoods of the complex root in the third quadrant.

The third example is a polynomial whose roots are all of multiplicity three. The roots are the five roots of unity, i.e.

$$p_3(z) = (z^5 - 1)^3. \tag{23}$$

The results are presented in Figs. 19–26. The following methods performed very well: Halley's method (Fig. 20 left), Dong's D2 (Figs. 21 right) and LCN2 (Fig. 24 right). Newton's scheme (Fig. 19 left) and LCN5 (Fig. 26 left) did not perform as well as the previously listed ones.The others show chaotic behavior.

The fourth example is a polynomial whose roots are all of multiplicity four. The roots are the seven roots of unity, i.e.

$$p_4(z) = (z^7 - 1)^4. \tag{24}$$

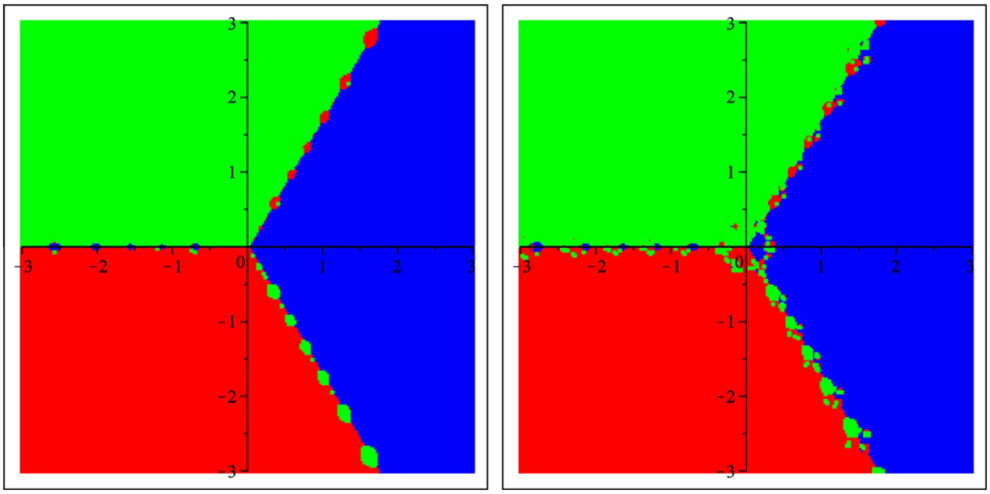

Fig. 46. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^3 - 1)^6$.

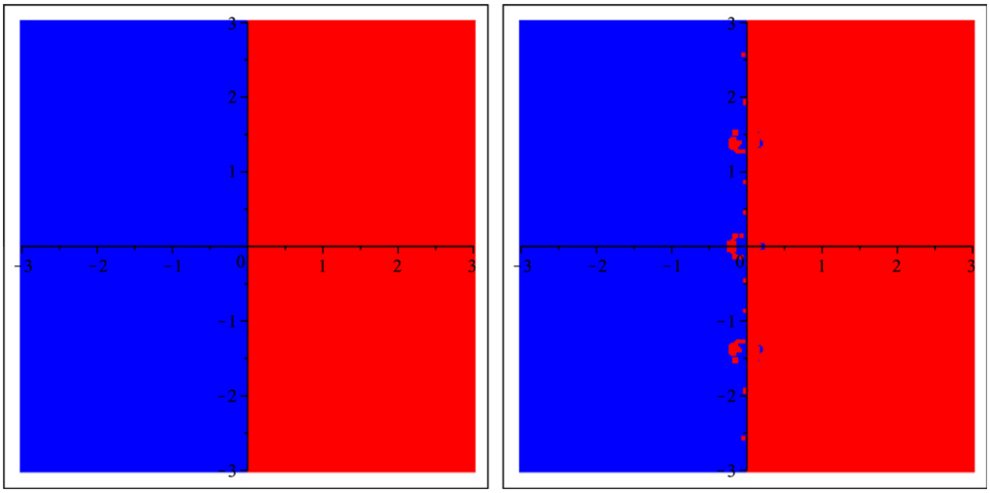

Fig. 47. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^n - 1)^3$ for $n = 2$.

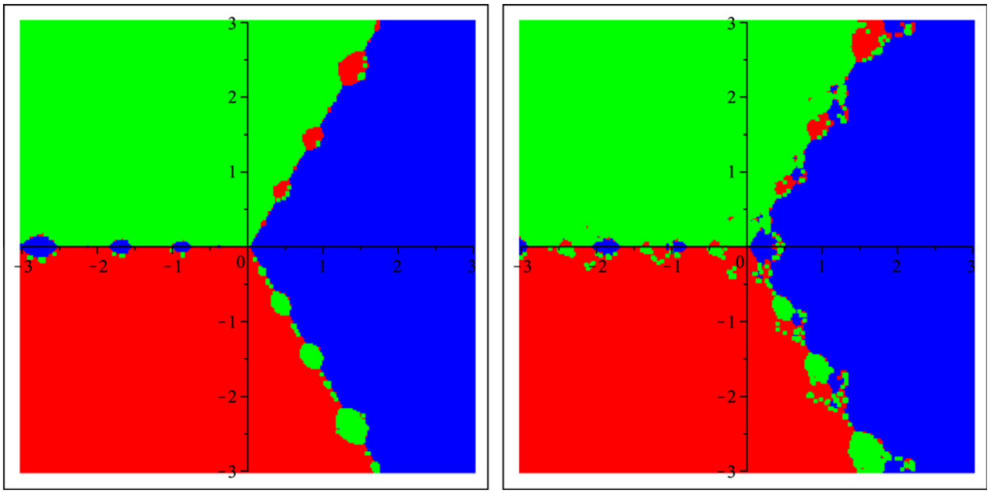

Fig. 48. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^n - 1)^3$ for $n = 3$.

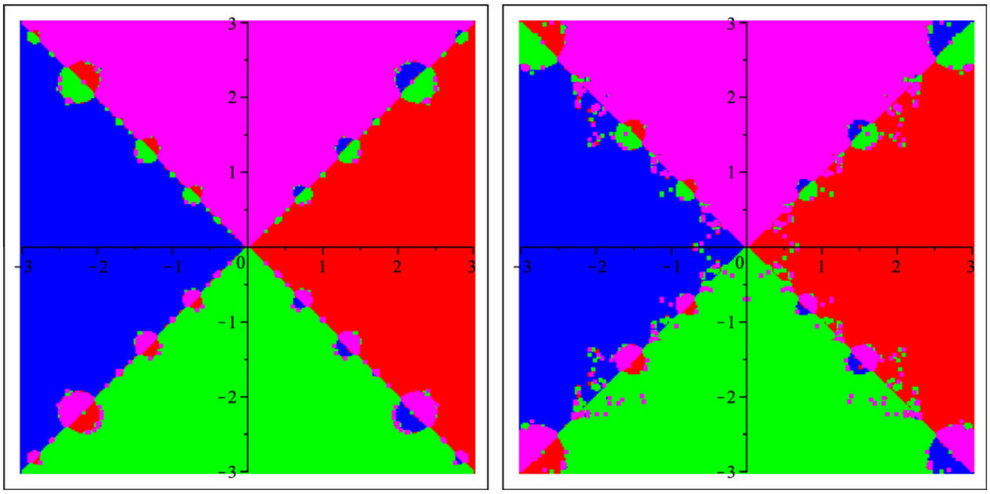

Fig. 49. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^n - 1)^3$ for $n = 4$.

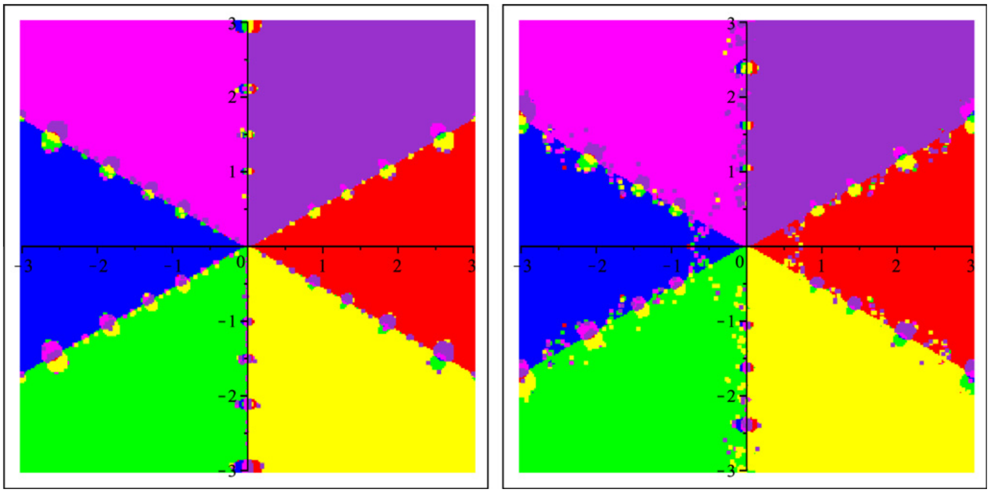

Fig. 50. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^n - 1)^3$ for $n = 6$.

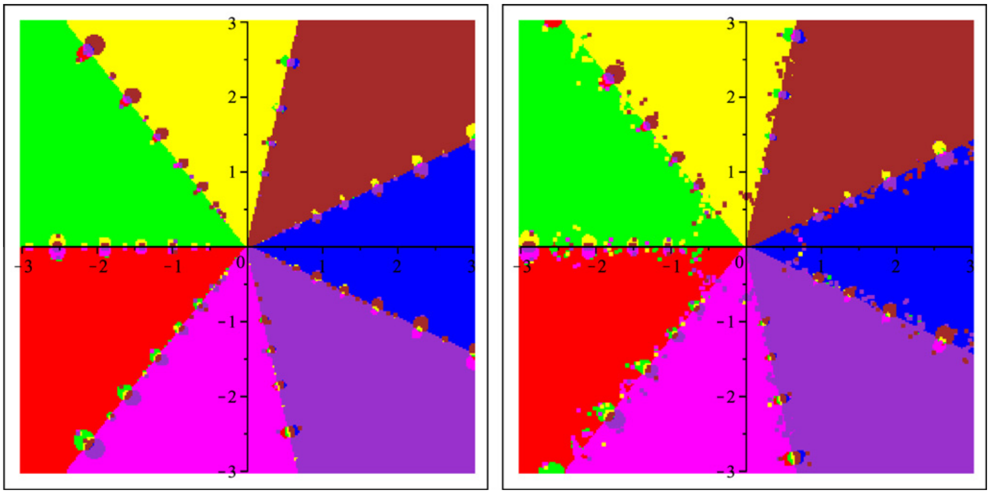

Fig. 51. Third order method due to Halley (left) and LCN2 fourth order scheme (right) for the polynomial $(z^n-1)^3$ for $n = 7$.

The results are presented in Figs. 27–34. Halley's method (Fig. 28 left) and the fourth order method LCN2 (Fig. 32 right) performed well, followed by LCN1 (Fig. 32 left) and LCN5 (Fig. 34 left). All the other schemes exhibit chaotic behavior.

In our next example we took the polynomial

$$p_5(z) = (z^4-1)^5, \tag{25}$$

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 results are shown in Figs. 35–42. The best methods are Dong's D2 (Fig. 37 right) and the fourth-order methods LCN1 and LCN2 (Fig. 39). All the other schemes exhibit chaotic behavior.

In order to summarize these results, we have attached a weight to the quality of the results obtained by each method. The weight of 1 is for the smallest Julia set and a weight of 4 for scheme with chaotic behavior. We then averaged those results to come up with the smallest value for the best method overall and the highest for the worst. These data is presented in Table 1. As one can see the best methods are Halley and LCN2, followed by Newton's and Dong's D2. The worst ones are Victory-Neta (19/5), NC (18/5) and D1 (17/5).

Since Halley's third order method and LCN2 fourth order scheme are best, we have decided to test these on the polynomial

$$p_6(z) = (z^3-1)^m \tag{26}$$

for various values of $m = 3, 4, 5, 6$. The results are given in Figs. 43–46, where we presented Halley's method on the left and LCN2 on the right. It is clear that both methods performed well for all values of m.

The next question is how they compare for the same multiplicity but with increasing order of the polynomial. For that, we have taken

$$P_n(z) = (z^n-1)^3, \quad n = 2, 3, \ldots, 7. \tag{27}$$

The case $n = 3$ was considered earlier (example 3). We plot the results in Figs. 47–51 and as before, Halley's method on the left and LCN2 on the right. These figures demonstrate the superiority of Halley's method over LCN2.

3. Conclusions

In this article, we have compared 17 methods of various orders ($p = 1.5$ to $p = 4$) for the approximation of multiple roots of different multiplicity ($2 \leqslant m \leqslant 6$). We have seen that the best methods are Halley's third order scheme and the fourth order method LCN2. The worst are Dong's D1, NC and Victory-Neta. We have then experimented with the top two schemes for polynomials having three roots with increasing multiplicity and polynomials of increasing degree and same multiplicity ($m = 3$). In the first case both methods performed well, but Halley's scheme was superior when the order of the polynomial has increased from $n = 3$ to $n = 7$.

Acknowledgement

Professor Chun's research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0025877).

References

[1] M. Scott, B. Neta, C. Chun, Basin attractors for various methods, Appl. Math. Comput. 218 (2011) 2584–2599, doi:10.1016/j.amc.2011.07.076.
[2] A.M. Ostrowski, Solutions of Equations and System of equations, Academic Press, New York, 1960.
[3] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice Hall, New Jersey, 1964.
[4] B. Neta, Numerical Methods for the Solution of Equations, Net-A-Sof, California, 1983.
[5] L.B. Rall, Convergence of the Newton process to multiple solutions, Numer. Math. 9 (1966) 23–37.
[6] E. Schröder, Über unendlich viele Algorithmen zur Auflösung der Gleichungen, Math. Ann. 2 (1870) 317–365.
[7] E. Hansen, M. Patrick, A family of root finding methods, Numer. Math. 27 (1977) 257–269.
[8] H.D. Victory, B. Neta, A higher order method for multiple zeros of nonlinear functions, Intern. J. Comput. Math. 12 (1983) 329–335.
[9] C. Dong, A basic theorem of constructing an iterative formula of the higher order for computing multiple roots of an equation, Math. Numer. Sin. 11 (1982) 445–450.
[10] C. Dong, A family of multipoint iterative functions for finding multiple roots of equations, Int. J. Comput. Math. 21 (1987) 363–367.
[11] B. Neta, A.N. Johnson, High order nonlinear solver for multiple roots, Computers and Mathematics with Applications, doi:10.1016/j.camwa.2007.09.001.
[12] B. Neta, Extension of Murakami's high-order non-linear solver to multiple root, Int. J. Comput. Math. 87 (2010) 1023–1031.
[13] S.G. Li, L.Z. Cheng, B. Neta, Some fourth-order nonlinear solvers with closed formulae for multiple roots, Comput. Math. Appl. 59 (2010) 126–135, doi:10.1016/j.camwa.2009.08.066.
[14] W. Werner, Iterationsverfahren höherer Ordnung zur Lösung nicht linearer Gleichungen, Z. Angew. Math. Mech. 61 (1981) T322–T324.
[15] B. Neta, New third order nonlinear solvers for multiple roots, Appl. Math. Comput. 202 (2008) 162–170, doi:10.1016/j.amc.2008.01.031.
[16] E. Halley, A new, exact and easy method of finding the roots of equations generally and that without any previous reduction, Phil. Trans. Roy. Soc. Lond. 18 (1694) 136–148.
[17] V.F. Candela, A. Marquina, Recurrence relations for rational cubic methods II: the Chebyshev method, Computing 45 (1990) 355–367.
[18] D.J. Hofsommer, Note on the computation of the zeros of functions satisfying a second order differential equation, Math. Table Other Aids Comput. 12 (1958) 58–60.
[19] D.B. Popovski, A family of one point iteration formulae for finding roots, Int. J. Comput. Math. 8 (1980) 85–88.
[20] N. Osada, An optimal multiple root-finding method of order three, J. Comput. Appl. Math. 51 (1994) 131–133.